



DEUTSCHES  
PATENT- UND  
MARKENAMT

12 Offenlegungsschrift  
DE 198 46 855 A 1

21 Aktenzeichen: 198 46 855.5  
22 Anmeldetag: 12. 10. 98  
43 Offenlegungstag: 6. 5. 99

51 Int. Cl.<sup>8</sup>:  
G 05 B 19/04  
G 06 F 11/00  
G 06 F 19/00  
G 01 R 31/00  
H 04 B 17/00  
H 04 M 1/24

DE 198 46 855 A 1

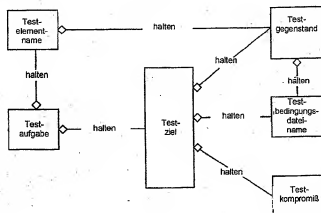
30 Unionspriorität:  
9-293329 10. 10. 97 JP  
71 Anmelder:  
Advantest Corp., Tokio/Tokyo, JP  
74 Vertreter:  
Betten & Resch, 80469 München

72 Erfinder:  
Umezu, Satoshi, Tokio/Tokyo, JP; Yamaguchi,  
Takahiro, Tokio/Tokyo, JP; Miyajima, Jun,  
Tokio/Tokyo, JP

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

54 Testsystem und Verfahren zum Steuern dieses Systems

51 Es wird ein Testsystem zum Durchführen einer Operation einer Anwendung offenbart, das eine Testausrüstung steuert von Testen einer zu testenden Vorrichtung durch Anzeigen von Bildern wie zum Beispiel Graphiksymbolen oder Druckknöpfen auf einem Bildschirm, und durch Auswählen dieser Bilder unter Verwendung einer Zeigevorrichtung (30), sowie ein Verfahren zum Steuern desselben. Ein Graphiksymbol, das der zu testenden Vorrichtung zugeordnet ist, sowie ein Graphiksymbol, das einem Testelement zugeordnet ist, werden angezeigt und anschließend verbunden und angezeigt. Außerdem wird ein Testparameter erzeugt, der dem jeweiligen Testelement zugeordnet ist, wobei ein Test ausgeführt wird durch Setzen entsprechender Testparameter, und wobei die erhaltenen Testdaten angezeigt werden.



DE 198 46 855 A 1

Die vorliegende Erfindung bezieht sich auf ein Testsystem, das eine zu testende Vorrichtung testet, indem es eine Teststeuervorrichtung und eine oder mehrere Testeinheiten verbindet, sowie auf ein zugehöriges Steuerverfahren. In dieser Beschreibung bedeutet eine zu testende Vorrichtung (device under test, DUT) eine Vorrichtung, ein System, eine Ausrüstung oder eine Komponente einer Ausrüstung, die durch das Testsystem gemäß der vorliegenden Erfindung getestet werden soll. Der Test umfaßt verschiedene Vorgänge, die für bekannte Testzwecke durchgeführt werden sollen, wie z. B. eine Herstellungsbewertung, eine Qualitätskontrolle, eine Korrektur, eine Kalibrierung, eine Ausrichtung, eine Einstellung, eine Leistungsbewertung, eine Diagnose und eine Produkteingangsspektion.

Weiterhin enthält eine "Testvorrichtung", wie sie hier verwendet wird, eine oder mehrere elektronische Ausrüstungen wie etwa einen Signalgenerator, einen Modulator, einen Demodulator, eine Eingabe/Ausgabe-Vorrichtung, einen Verstärker, einen Mischer, einen Codierer, einen Decodierer, ein Oszilloskop, ein Dehnungsmeßgerät, ein Wattmeter, ein Multimeter, einen Dämpfer, einen Detektor, einen Spektrumanalysator, einen Netzwerkanalysator, eine Halbleiter-Testvorrichtung, einen Synthesizer, einen Thermostaten oder eine Meßvorrichtung oder eine passive oder eine aktive Vorrichtung oder ein Instrument, das für die Ausführung einer Bewertung, eines Tests, einer Kalibrierung, einer Reparatur, einer Einstellung oder dergleichen für die Bewertung einer elektronischen Vorrichtung erforderlich ist.

Eine "Testaufgabe", wie sie hier verwendet wird, umfaßt einen Testgegenstand, eine Testbedingung, einen Testkompromiß, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt wird, ein Testelement und einen Test. Der Testgegenstand bezeichnet hier einen Gegenstand zum Testen einer DUT, wie z. B. die Oberwellenmessung. Eine Testbedingung, die beispielhaft anhand der oben erwähnten Oberwellenmessung beschrieben wird, umfaßt die Parameter des Testgegenstandes, wie z. B. eine Grundfrequenz und die Ordnung der zu messenden Oberwellen, sowie die Testparameter der im Test verwendeten Testelemente. Ferner umfaßt ein Testergebnis, das dem Testgegenstand entspricht, Daten, die erhalten werden durch Ausführen einer Operation entsprechend dem Testgegenstand mit den vom Testelement erhaltenen Testdaten. Hinsichtlich der oben erwähnten Oberwellenmessung als Beispiel ist z. B. ein Leistungsverhältnis zwischen der Grundwelle und den Oberwellen das Testergebnis, das dem Testgegenstand zugeordnet ist.

Außerdem umfaßt ein "Programmelement", wie es hier verwendet wird, die oben erwähnten Testaufgaben und die Bedingungen zum Steuern der Ausführungsreihenfolge der Testaufgaben wie z. B. Anfang, Ende, Pause, Warten, Warten auf das Spezifizieren eines Startzeitpunktes, eine Schleife zum Kombinieren derselben und eine Bedingungs-schleife.

Außerdem umfaßt ein "Speicher", wie er hier verwendet wird, einen RAM, einen ROM oder einen EPROM, eine Festspeichervorrichtung wie z. B. ein Diskettenlaufwerk, ein Festplattenlaufwerk und ein CD-ROM-Laufwerk, sowie irgendeine andere Speichereinrichtung, die im Stand der Technik bekannt ist.

Es ist erforderlich, mehrere Testvorrichtungen wie z. B. einen Spektrumanalysierer oder einen Netzwerkanalysierer zu verwenden, um ein Funktelefonssystem (PHS) oder ein Zellsystem zu testen. In einer solchen Testumgebung ist ein Benutzer des Testsystems nicht mit allen Testvorrichtungen vertraut.

Ein herkömmliches Testsystem verwendet einen Ansatz zur Visualisierung eines Testprogramms, um eine Umgebung zu schaffen, die einer Person, die mit der Testvorrichtung nicht vertraut ist, ermöglicht, das Testprogramm zu verwenden. Der Ausdruck Visualisierung bezieht sich auf ein Schema, das ein Programm als eine Sammlung von Piktogrammen darstellt, auch als Graphiksymbole bezeichnet, die jeweils einen Programmabschnitt darstellen, der die Aufnahme oder die Verarbeitung von Daten durchführt oder einen Graphen anzeigt, und spezifiziert eine Reihenfolge der Programmausführung durch Verbinden der angezeigten Graphiksymbole unter Verwendung einer Zeigevorrichtung mit einem Druckknopf auf einem Bildschirm eines Computers als Testvorrichtung.

Ein Benutzer des Testsystems wählt diese Graphiksymbole, die in einem Menü oder einem Werkzeugkasten gespeichert sind, aus durch Zeigen (drücken des Druckknopfes der Zeigevorrichtung und loslassen des Druckknopfes, ohne die Zeigevorrichtung zu bewegen) auf ein gewünschtes Graphiksymbol mit der Zeigevorrichtung mit Druckknopf. Anschließend wird das Zeigen erneut auf einem Fensterbildschirm zum Programmieren durchgeführt, was das ausgewählte Graphiksymbol auf dem Bildschirm zu platzieren. Jedem Graphiksymbol sind mehrere vorgegebene Verbindungspunkte zugeordnet, wobei ein Verbinden erreicht wird durch Zeigen auf einen Verbindungspunkt des Graphiksymbols, der einem Verbindungsstartpunkt entspricht, und einem Verbindungspunkt eines Graphiksymbols, der einem Verbindungsende entspricht, mit der Zeigevorrichtung mit Druckknopf.

#### (a) Erhalten der Testsystemstartoperation aus der Auswahl des Zwecks/Ziels

Wie oben beschrieben worden ist, verwendet ein Programmierverfahren eines herkömmlichen Testsystems einen Lösungsansatz zum Verbinden von Programmabschnitten, die die Datenaufnahme, eine arithmetische Operation oder eine Graphikdarstellung durchführen, durch Zeigen auf Punkte, die für entsprechende Graphiksymbole vorgegeben sind, und durch Verbinden derselben. Ein solcher Verbindungsansatz weist das Problem auf, daß der Benutzer das System nicht effektiv nutzen kann, sofern er nicht die Funktion jedes Graphiksymbols und die Funktion jedes Verbindungspunktes vollständig verstanden hat. Daher konzentriert der Benutzer des Testsystems viel Aufwand auf ein Programmierverfahren "wie sind die vom Testsystem zur Verfügung gestellten Programmabschnitte zu kombinieren", statt ein Verfahren zum Bewerten einer DUT zu betrachten. Insbesondere wenn beachtigt ist, einen geeigneten Test in einer Umgebung durchzuführen, in der eine DUT nicht getestet werden kann, sofern nicht mehrere Testvorrichtungen verwendet werden, ist ein Testsystem erforderlich, das dem Benutzer ermöglicht, seine Aufmerksamkeit auf die genaue Bewertung der DUT zu richten, selbst wenn er das Testsystem nicht vollständig verstanden hat.

Wenn ein Testsystem vorliegt, das der Benutzer starten kann, indem er ein Objekt oder ein Ziel des Tests auswählt, kann er seine Aufmerksamkeit darauf richten, wie die DUT bewertet wird. Es ist somit erwünscht, eine Testvorrichtung zu erreichen, die durch Auswählen eines Zwecks oder eines Zieles eines Tests betreiben werden kann.

#### (b) Erhalten eines Testsystems, auf dem ein Hochgeschwindigkeitstest oder ein hochgenauer Test spezifiziert werden kann

Beim Nutzen einer Testvorrichtung oder eines Testsy-

stems ist es notwendig, alternativ auszuwählen, ob der Test mit einer hohen Geschwindigkeit oder mit einer hohen Genauigkeit durchgeführt werden soll. Es besteht ein Kompromiß zwischen einer hohen Testgeschwindigkeit und einer hohen Testgenauigkeit. Das heißt, wenn der Test mit einer hohen Geschwindigkeit durchgeführt wird, muß die Testgenauigkeit geopfert werden. Wenn im Gegensatz hierzu der Test mit einer hohen Genauigkeit durchgeführt wird, ist eine längere Zeitspanne erforderlich.

Ein solcher Kompromiß zwischen einem Hochgeschwindigkeitstest und einem hochgenauen Test war jedoch in herkömmlichen Testsystemen nicht klar. Der Benutzer kann somit keine genaue Einstellung entsprechend dem Kompromiß erreichen, "wie ist die Testvorrichtung zu verwenden", sofern er nicht den Test mit mehreren Versuchen und Fehlversuchen durchführt. Außerdem besteht das Problem, daß selbst ein solcher Versuch-Irrtum-Test nicht durchgeführt werden kann, sofern der Benutzer nicht die Hardware-Konfiguration der Testvorrichtung und die entsprechenden Testparameter vollständig verstanden hat.

Es ist somit ferner erwünscht, eine Testvorrichtung zu schaffen, die es ermöglicht, den Kompromiß eines schnellen/hochgenauen Tests selbst dann zu spezifizieren, wenn der Benutzer die genaue interne Konfiguration des Testsystems oder der Vorrichtung und die Testparameter nicht vollständig verstanden hat.

(c) Erhalten eines Testsystems, bei dem dann, wenn bestimmte Koordinaten auf einer Graphikdarstellung als Testergebnis spezifiziert werden, ein Dienst entsprechend dem spezifizierten Bereich gestartet wird

Es wird gefordert, daß das Testsystem Testdaten einer DUT in Form eines Graphen darstellt, der es ermöglicht, z. B. einen Leistungswert einer unerwarteten Störung (spurious) schnell und mit hoher Genauigkeit zu erhalten. Wenn jedoch z. B. beabsichtigt ist, einen Leistungswert einer Störung mit dem herkömmlichen Testsystem zu erhalten, ist es erforderlich, eine Programmierung und eine Operation erneut durchzuführen, wie z. B. (1) das Beenden des Tests, (2) das Erstellen eines Programms zum Hinzufügen von Testdaten, und (3) das erneute Durchführen des Tests, um den Leistungswert zu finden. Wenn in diesem Fall die Störung, von der die Leistung ermittelt wird, ein Teil der Testdaten ist, ist es erforderlich, Matrixdaten in den entsprechenden Testdaten aus den Positionskordinaten der Störung auf der graphischen Darstellung zu lokalisieren und ein Programm hinzuzufügen, das die lokalisierten Matrixdaten hinzufügt. Somit erfordert das herkömmliche Testsystem sehr viel Aufwand, nur um den Leistungswert einer Störung auf der graphischen Darstellung zu finden.

Es ist daher bequem, wenn ein Testsystem vorliegt, bei dem dann, wenn bestimmte Koordinaten auf einer graphischen Darstellung als ein Testergebnis spezifiziert werden, ein Dienst entsprechend dem spezifizierten Bereich gestartet wird.

(d) Erhalten eines Testsystems bei dem ein neuer Testparameter für ein Testelement aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Ergebnisses erzeugt werden kann, das vom Testelement erhalten wird, und bei dem ein neuer Test mit dem erzeugten Testparameter gestartet wird

Wenn eine Frequenz einer unerwarteten Störung wie oben beschrieben genauer analysiert wird, wird ein erneutes Testen unter Verwendung eines Frequenzbereiches, der die Störung enthält, als einen neuen Testparameter erforderlich.

Insbesondere im Fall einer Störung oder dergleichen entsteht das Problem bezüglich des Pegels oder der Frequenz der Störung für einen Signalpegel, der der Hauptgegenstand des Tests ist. Somit ist eine Umgebung erforderlich, in der der Signalpegel, der der Hauptgegenstand des Tests ist, und die Störung gleichzeitig angezeigt und verglichen werden können.

Auch das herkömmliche Testsystem kann eine Umgebung implementieren, in der ein Testparameter erneut eingestellt werden kann, indem eine Instrumenten tafel angezeigt wird und ein Programm zum Steuern der Hardware für die Mittenfrequenz und eine Frequenzspanne erzeugt wird. Um jedoch mehrere unerwartete Störungen zu analysieren, muß der Benutzer des Testsystems das Programm modifizieren und die Testparameter so einstellen, daß die Mittenfrequenz und die Frequenzspanne auf der Instrumenten tafel angezeigt werden, so daß eine Instrumenten tafel zum Steuern der neuen Hardware erneut angezeigt werden kann.

Wenn ein Mechanismus vorhanden ist, mit dem dann, wenn eine Störung auf der graphischen Darstellung gefunden wird, ein neuer Testparameter erzeugt werden kann, indem der Bereich der gefundenen Störung auf der graphischen Darstellung spezifiziert wird, muß der Benutzer der Testvorrichtung das Testprogramm nicht jedes mal dann modifizieren, wenn eine Störung gefunden wird. Wenn außerdem der automatisch erzeugte Testparameter automatisch auf ein Instrument zum Analysieren der Störung gesetzt wird, muß der Benutzer des Testsystems nicht die Frequenz der Störung oder dergleichen aus der graphischen Darstellung ablesen und den Testparameter auf der neu erzeugten Instrumenten tafel einstellen. Diese Funktionen können die Belastung des Benutzers des Testsystems deutlich reduzieren. Der Benutzer des Testsystems kann sich folglich auf die Analyse der Störung konzentrieren, ohne durch die Modifizierung des Programms abgelenkt zu sein. Das herkömmliche Testsystem bietet jedoch keine solche Funktion.

Es ist daher erwünscht, ein Testsystem zu schaffen, das einen neuen Testparameter eines Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung des vom Testelement erhaltenen Ergebnisses erzeugen kann, ohne ein Programm zu programmieren, und einen neuen Test fit dem erzeugten Testparameter starten kann.

(e) Erhalten eines Testsystems, bei dem ein neuer Testparameter für ein weiteres Testelement aus einem spezifizierten Bereich auf einer graphischen Darstellung eines von einem Testelement erhaltenen Ergebnisses erzeugt werden kann und ein neuer Test für ein weiteres Testelement mit dem erzeugten Testparameter gestartet werden kann

Um die Analyse einer Störung wie oben beschrieben durchzuführen, ist es erforderlich, eine Testausrüstung wie z. B. einen Spektrumanalysierer, einen Frequenzzähler oder einen Leistungsmesser zur Verfügung zu stellen. Wenn die Bewertung mit der geeigneten Testausrüstung erreicht werden kann, während diese Testausrüstung mit dem Testsystem verbunden ist, kann die Analyse mit einer erforderlichen Testgenauigkeit effizient durchgeführt werden. Wenn die Störung während eines Tests mit einem Spektrumanalysierer erfaßt wird, kann dann, wenn ein weiterer Test unter Verwendung eines Frequenzzählers aus einem Bereich auf der graphischen Darstellung für die zu analysierende Störung durchgeführt wird, die Frequenz der Störung mit hoher Genauigkeit getestet werden.

Wenn die Störung auf der graphischen Darstellung festgestellt wird, kann dann, wenn ein Testparameter zum Analysieren der Störung automatisch aus einem spezifizierten Bereich auf der graphischen Darstellung erzeugt wird und au-

tionatisch ein Testelement zum Analysieren der Störung eingesetzt wird, der Benutzer des Testsystems nicht die Frequenz der Störung oder dergleichen aus der graphischen Darstellung ablesen, auf der die Störung zu finden ist, und braucht den Testparameter auf der neu erzeugten Instrumententafel nicht manuell einstellen. Folglich kann der Benutzer des Testsystems seine Aufmerksamkeit auf die Analyse der Störung konzentrieren, ohne durch die Modifizierung des Programms abgelenkt zu sein. Das herkömmliche Testsystem bietet jedoch keine solche Funktion.

Es ist daher erwünscht, ein Testsystem zu schaffen, das einen neuen Testparameter eines weiteren Testelements erzeugen kann, ohne ein Programm umzuprogrammieren, und einen neuen Test entsprechend einem weiteren Testelement mit dem erzeugten Testparameter starten kann.

Es ist die erste Aufgabe der vorliegenden Erfindung, ein Testsystem zu schaffen, das eine Operation starten kann, indem ein Objekt oder ein Ziel eines Tests ausgewählt wird. Es ist eine zweite Aufgabe der vorliegenden Erfindung, ein Testsystem zu schaffen, das spezifizieren kann, ob ein Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt wird. Es ist eine dritte Aufgabe der vorliegenden Erfindung, ein Testsystem zu schaffen, das einen Dienst starten kann, der für einen spezifizierten Bereich ausgelegt ist, ohne ein Programm umzuprogrammieren, wenn bestimmte Koordinaten auf einer graphischen Darstellung als ein Testergebnis spezifiziert werden. Es ist eine vierte Aufgabe der vorliegenden Erfindung, ein Testsystem zu schaffen, das einen neuen Testparameter eines Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung des vom Testelement erhaltenen Ergebnisses erzeugen kann, ohne ein Programm umzuprogrammieren, und einen neuen Test mit dem erzeugten Testparameter starten kann. Es ist eine fünfte Aufgabe der vorliegenden Erfindung, ein Testsystem, das einen neuen Testparameter eines weiteren Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung des vom Testelement erhaltenen Ergebnisses ohne Umprogrammierung eines Programms erzeugen kann und einen neuen Test mit dem erzeugten Testparameter starten kann, sowie ein Steuerverfahren hierfür zu schaffen.

Diese Aufgaben werden erfindungsgemäß gelöst durch ein Testsystem und ein Verfahren, die die in den unabhängigen Ansprüchen 1 und 5 angegebenen Merkmale besitzen. Die abhängigen Ansprüche sind auf bevorzugte Ausführungsformen gerichtet.

In einer bevorzugten Ausführungsform umfaßt das Testsystem der vorliegenden Erfindung eine Testaufgabeneinrichtung, eine Testparameterspeichereinrichtung, eine Testdatenspeichereinrichtung, eine virtuelle Instrumenteneinrichtung, eine Darstellungsblatteinrichtung und eine Testklasseneinrichtung, wobei die Operation gestartet werden kann durch Zurückverfolgen ausgehend von einem Test-Zweck/Ziel, und der Kompromiß eingestellt werden kann, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt wird, indem lediglich ein Test-Zweck/Ziel spezifiziert wird.

Außerdem kann eine Funktion erhalten werden, die einen Dienst starten kann, der für einen spezifizierten Bereich ausgelegt ist, wenn bestimmte Koordinaten auf einer graphischen Darstellung als ein Testergebnis spezifiziert sind, indem eine Schlitzinformationseinrichtung und eine virtuelle Instrumentendiensteinrichtung der obigen Kombination hinzugefügt werden. Wenn eine Schlitzinformationseinrichtung und eine Wechselwirkungseinrichtung der obigen Kombination hinzugefügt werden, können alternativ Funktionen erreicht werden (a) zum Erzeugen eines neuen Testparameters für ein Testelement aus einem spezifizierten Bereich auf einer

graphischen Darstellung, die ein vom Testelement erhaltenes Ergebnis ist, ohne Umprogrammierung eines Programms, und Starten eines neuen Tests mit dem erzeugten Testparameter; und (b) zum Erzeugen eines neuen Testparameters für ein Testelement aus einem spezifizierten Bereich auf einer graphischen Darstellung, die ein von einem weiteren Testelement erhaltenes Ergebnis ist, und Starten eines neuen Tests mit dem erzeugten Testparameter.

Wenn der obigen Kombination eine Testplaneinrichtung hinzugefügt wird, wird es außerdem möglich, ein Programmelement oder ein zusammengesetztes Programmelement zu kombinieren, wodurch das Schreiben und Lauflassen eines Programms möglich wird, das eine Reihenfolge für die Durchführung von Testaufgaben oder die Steuerung der Reihenfolge der Ausführung einstellen kann.

Weitere Merkmale und Vorteile der vorliegenden Erfindung werden deutlich beim Lesen der folgenden Beschreibung bevorzugter Ausführungsformen, die auf die beigefügten Zeichnungen Bezug nimmt; es zeigen:

**Fig. 1** ein Schaubild, das die Vererbung von Objekten zeigt;

**Fig. 2** ein Schaubild, das eine Speicherkonfiguration für ein Basisobjekt und ein abgeleitetes Objekt zeigt;

**Fig. 3** ein Schaubild, das ein Beispiel einer Darstellung des Zuordnens von Objekten zeigt;

**Fig. 4** ein Schaubild, das eine Speicherkonfiguration für ein Zuordnungsobjekt und ein zugeordnetes Objekt zeigt;

**Fig. 5** ein Schaubild, das eine Konfiguration eines Basisobjektes zeigt, das zum Implementieren der Testaufgabeneinrichtung erforderlich ist;

**Fig. 6** ein Flußdiagramm, das die Schritte eines Prozesses zeigt, der von der Testaufgabeneinrichtung ausgeführt wird;

**Fig. 7** ein Flußdiagramm, das die Schritte eines Prozesses zeigt, der von der Testaufgabeneinrichtung ausgeführt wird;

**Fig. 8** ein Blockschaltbild, das die Konfiguration von Basisobjekten zur Implementierung der Testparameterspeichereinrichtung, der Testdatenspeichereinrichtung, der Schlitzinformationseinrichtung, der Darstellungsblatteinrichtung, der Wechselwirkungseinrichtung, der virtuellen Instrumentendiensteinrichtung und der virtuellen Instrumenteneinrichtung zeigt;

**Fig. 9** ein Schaubild, das eine Konfiguration eines Basisobjekts zeigt, das zur Implementierung der Testklasseneinrichtung erforderlich ist;

**Fig. 10** ein Schaubild, das eine Konfiguration eines Basisobjektes zeigt, das zum Implementieren der Testplaneinrichtung erforderlich ist;

**Fig. 11** ein Flußdiagramm, das die Schritte eines Prozesses zeigt, der von der Testplaneinrichtung ausgeführt wird;

**Fig. 12** ein Blockschaltbild, das eine Konfiguration des Testsystems gemäß einer Ausführungsform der Erfindung zeigt;

**Fig. 13** ein Flußdiagramm, das die Operationsprozedur eines Ereignis-Erfassungs/Informierungs-Prozesses eines Ereignis-Erfassungs/Informierungs-Abschnitts zeigt;

**Fig. 14** ein Beispiel einer GUI-Implementierung in einem Testsystem, das die Operation startet durch Auswählen eines Test-Zwecks/Ziels;

**Fig. 15** ein Flußdiagramm, das eine Operationsprozedur beim Starten eines Tests durch Spezifizieren eines Testzieles zeigt;

**Fig. 16** ein Flußdiagramm, das eine Operationsprozedur beim Anzeigen von Testdaten als ein Ergebnis nach der Durchführung eines Tests zeigt;

**Fig. 17** ein Flußdiagramm, das eine Operationsprozedur beim Anzeigen von Testdaten als Ergebnis nach der Durchführung eines Tests zeigt;

**Fig. 18** ein Beispiel einer GUI-Implementierung in einem

Testsystem, das einen entsprechenden Dienst startet, wenn bestimmte Koordinaten auf einer graphischen Darstellung eines Testergebnisses spezifiziert werden;

Fig. 19 ein Flußdiagramm, das eine Operationsprozedur zum Durchführen eines Dienstes zum Erhalten eines Spitzenwertes in einem spezifizierten Bereich zeigt;

Fig. 20 ein Beispiel einer GUI-Implementierung zum Erzeugen eines neuen Testparameters aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Testergebnisses und zum Starten des Tests;

Fig. 21 ein Beispiel einer GUI-Implementierung zum Erzeugen eines neuen Testparameters aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Testergebnisses und zum Starten eines neuen Tests;

Fig. 22 ein Beispiel einer GUI-Implementierung zum Erzeugen eines neuen Testparameters aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Testergebnisses und zum Starten eines neuen Tests;

Fig. 23 ein Beispiel einer GUI-Implementierung zum Erzeugen und Abarbeiten eines Programms;

Fig. 24 ein Flußdiagramm, das eine Operationsprozedur beim Erzeugen eines Programms zeigt; und

Fig. 25 ein Flußdiagramm, das eine Operationsprozedur beim Abarbeiten eines Programms zeigt.

Vor der genaueren Beschreibung der vorliegenden Erfindung wird das Konzept der Aufgabe beschrieben.

#### Vererbung eines Objekts

Ein Objekt ist ein Attribut (Daten), das in einen Prozeß integriert ist und in einem Speicher gespeichert ist. Die Fig. 1 ist ein Schaubild, das die Vererbung von Objekten zeigt. Außerdem ist die Fig. 2 ein Schaubild, das eine Konfiguration von Objekten in einem Speicher zeigt. Wie in Fig. 1 gezeigt, ist ein Pfeil, der mit "Vererbung" bezeichnet ist, von einem abgeleiteten Objekt zu einem Basisobjekt gezeichnet. Dies zeigt an, daß das abgeleitete Objekt ein vom Basisobjekt vererbtes Objekt ist. Die Fig. 2 zeigt eine Konfiguration des Basisobjektes und des abgeleiteten Objektes in einer solchen Beziehung im Speicher.

Ein abgeleitetes Objekt besitzt an seiner Spitze ein Basisobjekt. Ein Einsprungpunkt eines abgeleiteten Objektes zeigt somit auf die gleiche Adresse wie der Einsprungpunkt eines Basisobjektes, das im abgeleiteten Objekt enthalten ist. Das heißt, ein abgeleitetes Objekt kann als ein Basisobjekt behandelt werden. Wie beschrieben worden ist, wird die Tatsache, daß ein abgeleitetes Objekt an der Spitze eines Speichers ein Basisobjekt besitzt, ausgedrückt als ein abgeleitetes Objekt, das ein Basisobjekt beerbt. Wenn ein abgeleitetes Objekt ein Basisobjekt beerbt, wird damit außerdem ausgedrückt, daß das Basisobjekt als ein Stammobjekt bezeichnet wird und das abgeleitete Objekt als Abkömmlingsobjekt bezeichnet wird.

In einem in Fig. 2 gezeigten Beispiel sind zwei Prozesse 1 in einem abgeleiteten Objekt implementiert. "Prozeß 1" nahe einem Einsprungpunkt des abgeleiteten Objekts im Speicher ist ein durch ein Basisobjekt definierter Prozeß. Ein weiterer Prozeß 1 ist ein durch das abgeleitete Objekt definierter Prozeß.

Wenn "Prozeß 1" für das abgeleitete Objekt angefordert wird, wird der vom abgeleiteten Objekt definierte Prozeß 1 in der folgenden Prozedur ausgeführt.

- Die Ausführung eines Prozeß 1 wird für ein Objekt angefordert;
- Das Objekt sucht den angeforderten Prozeß von einem Abkömmling-Objekt zu einem Stamm-Objekt; und

c) Das Objekt erfaßt einen Prozeßnamen, gibt einen Wert und ein Argument zurück, die zu denjenigen des angeforderten Prozesses passen, und führt diesen aus. Selbst wenn somit das abgeleitete Objekt als ein Basisobjekt behandelt wird, wird der vom abgeleiteten Objekt definierte Prozeß 1 ausgeführt, wenn die Ausführung des Prozeß 1 angefordert wird.

Selbst für ein Abkömmling-Objekt jedoch kann ein Stamm-Objekt durch Spezifizieren des Stamm-Objekts ausgeführt werden. Wenn außerdem der Prozeß 1 nicht in einem Stamm-Basisobjekt implementiert ist, kann die Ausführung des Prozeß 1 nicht für ein Objekt angefordert werden, das als ein Basisobjekt behandelt wird. Nur ein im Basisobjekt definierter Prozeß kann die Ausführung des Basisobjektes anfordern.

Ein abgeleitetes Objekt kann auf den Inhalt eines "Basisobjektattributs 1", den ein Basisobjekt besitzt, zugreifen oder diesen modifizieren.

#### Zuordnen des Objekts

Die Fig. 3 ist ein Schaubild, das ein Beispiel der Darstellung des Zuordnens von Objekten zeigt. In Fig. 3 ist eine Linie, die mit "zuordnen" bezeichnet ist, von einem "zuordnenden Objekt" zu einem "zugeordneten Objekt" gezeichnet, wobei am zuordnenden Objekt eine Raute gezeigt ist. Dies stellt dar, daß das zuordnende Objekt das zugeordnete Objekt hält. Die Fig. 4 zeigt die Konfiguration eines solchen zuordnenden Objekts und eines zugeordneten Objekts in einem Speicher.

Wenn das zuordnende Objekt nur ein zugeordnetes Objekt besitzt, ist ein Einsprungpunkt des zugeordneten Objekts im Speicher gespeichert im Speicher des zuordnenden Objekts. Wenn außerdem das zuordnende Objekt mehrere zugeordnete Objekte besitzt, ist ein Satz von Einsprungpunkten (Matrixdaten) der zugeordneten Objekte im Speicher im Speicherbereich des zuordnenden Objekts gespeichert. Mit dieser Struktur kann das zuordnende Objekt die Einsprungpunkte im Speicher des zugeordneten Objekts erhalten durch Zugreifen auf den Speicher, der das zuordnende Objekt verwaltet, und kann einen Prozeß starten, den das zugeordnete Objekt besitzt.

Außerdem kann selbst ein Abkömmling-Objekt des zugeordneten Objekts als ein zugeordnetes Objekt eines Stamms gehalten werden. Ein Abkömmling-Objekt des zuordnenden Objekts kann jedoch auf ein in einem zuordnenden Objekt zugeordnetes Objekt zugreifen oder ein Objekt modifizieren, das es zuordnet.

Im folgenden wird die vorliegende Erfindung genauer beschrieben.

#### [Testaufgabeneinrichtung]

Da das herkömmliche Testsystem kein Test-Zweck/Ziel auswählen kann, ist viel Aufwand bei der Implementierung eines Programms erforderlich, so daß die Aufmerksamkeit nicht auf eine Einzelheit der Bewertung einer DUT gerichtet werden kann. Außerdem kann der Benutzer in herkömmlichen Testsystemen eine genaue Einstellung für das Testsystem finden, die den Kompromiß, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt werden soll, umfaßt, nur durchführen, nachdem er mehrere Versuche durchgeführt hat. Um diese Probleme zu lösen, schafft die vorliegende Erfindung eine neuartige Testaufgabeneinrichtung.

Die Testaufgabeneinrichtung dient zum Schaffen eines GUI zum Definieren eines Test-Zwecks/Ziels. Die Testauf-

gabeneinrichtung speichert ein Testelement, das zur Durchführung eines Tests spezifiziert ist, und wenigstens ein Testziel. Das Testziel enthält einen Testgegenstand, eine Testbedingung, den Kompromiß, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt werden soll, und dergleichen. Als Antwort auf eine Anforderung vom Benutzer des Testsystems wird anschließend ein mit wenigstens einem Testgegenstand spezifiziertes Testziel zur virtuellen Instrumenteneinrichtung gesendet, wo die Messung gestartet wird.

Die Fig. 5 ist ein Schaubild, das eine Konfiguration eines Basisobjektes zeigt, das zur Implementierung der Testaufgabeneinrichtung erforderlich ist. Wie in Fig. 5 gezeigt, enthalten die Basisobjekte, die die Testaufgabeneinrichtung implementieren, einen "Testgegenstand" einen "Testbedingungsdateinamen", einen "Testkompromiß", ein "Testziel", einen "Testelementnamen" und eine "Testaufgabe". Im folgenden wird jedes Objekt beschrieben.

#### Testgegenstand

Dies ist ein Objekt zum Speichern eines Testgegenstandsnamens im Speicher als Attribut. Der Testgegenstand enthält einen Testbedingungsdateinamen, der im Speicher eine Testbedingung speichert, die der Benutzer des Testsystems in einem Testgegenstand verwenden kann, wenn er den Testgegenstand spezifiziert. Der Testgegenstand hat die Funktion zum Anzeigen eines Testgegenstand-Graphiksymbols.

#### Testbedingungsdateiname

Dies ist ein Objekt zum Speichern eines Namens der Testbedingungen im Speicher als Attribut. Der Testbedingungsdateiname hat die Funktion zum Anzeigen eines Testbedingungsdateiname-Graphiksymbols.

#### Testkompromiß

Dies ist ein Objekt zum Speichern der Testkompromißinformationen darüber, ob der Test mit einer hohen Geschwindigkeit oder mit hoher Genauigkeit ausgeführt werden soll, im Speicher als Attribut. Der Testkompromiß hat die Funktion zum Anzeigen eines Testelement-Graphiksymbols, das einem Hochgeschwindigkeitstest oder einem hochgenauen Test zugeordnet ist, durch Zugriff auf einen Testelementnamen, den die Testaufgabe speichert.

#### Testziel

Dies ist ein Objekt zum Zuordnen einer Testbedingung, eines Testkompromisses und eines Testgegenstands.

#### Testelementname

Dies ist ein Objekt zum Speichern eines Namens des Testelements, das im Test verwendet wird, im Speicher als Attribut. Außerdem ordnet er mehrere Testgegenstände zu, die in einem Testelement verwendet werden können. Zum Beispiel existiert ferner ein Testelement wie z. B. ein Frequenzzähler, dessen Testgegenstand durch einen Vorgabewert mit nur einem Testelement definiert ist. Durch Darstellen eines solchen Testelements speichert der Testelementname einen Testgegenstand, der als Vorgabe gestartet wird, im Speicher als Attribut. Der Testelementname hat die Funktion des Anzeigens eines Graphiksymbols, das ein Testelement darstellt.

#### Testaufgabe

Die Testaufgabe ist ein Objekt zum Zuordnen mehrerer Testziele und eines Testelementnamens. Die Testaufgabe ist eines der Programmelemente und besitzt einen "Ausführungs"-Prozeß. Der Ausführungs-Prozeß der Testaufgabe startet den Test durch Senden eines Testziels, das ein abgeleitetes Objekt des "virtuellen Instruments" besitzt, das dem Testelementnamen entspricht, den es zuordnet.

Die Testaufgabe ordnet mehrere Testziele und einen Testelementnamen zu, wodurch eine GUI zum Definieren eines Test-Zwecks/Ziels geschaffen wird.

Die Fig. 6 und 7 zeigen ein Flußdiagramm, das die Schritte eines Prozesses zeigt, der von der Testaufgabeneinrichtung ausgeführt wird. Hierbei erzeugt die Testaufgabeneinrichtung einen ersten Anzeigebereich, der auf der graphischen Anzeigevorrichtung überlappend angezeigt werden kann.

Zuerst zeigt die Testaufgabeneinrichtung wenigstens ein Graphiksymbol, das den Testelementnamen darstellt, im ersten Anzeigebereich (Schritt a1), und ferner ein Graphiksymbol, das eine DUT darstellt, auf dem ersten Anzeigebereich (Schritt a2) an.

In diesem Zustand wählt der Benutzer des Testsystems einen Testelementnamen unter Verwendung der Zeigevorrichtung aus (Schritt a3). Die Testaufgabeneinrichtung setzt den ausgewählten Testelementnamen in die Zuordnungsbeziehung der Testaufgabe (Schritt a4) und zeigt Graphiksymbole der Testelementnamen an, die dem Hochgeschwindigkeitstest und dem hochgenauen Test zugeordnet sind (Schritt a5). Anschließend zeigt die Testaufgabeneinrichtung im ersten Anzeigebereich ein Graphiksymbol an, das einen Testgegenstand darstellt, dem der Testelementname zugeordnet (Schritt a6).

Die Testaufgabeneinrichtung ermittelt in dieser Anzeigeaktion, ob der Benutzer des Testsystems das Graphiksymbol des Testgegenstands auswählt (Schritt a7), und bewegt das Graphiksymbol, das den ausgewählten Testgegenstand darstellt, falls ausgewählt, zu einer Position im ersten Anzeigebereich, an der das ausgewählte Graphiksymbol angezeigt wird (Schritt a8) und setzt den ausgewählten Testgegenstand im Testziel, das die Testaufgabe besitzt, so daß das Testziel diese zuordnet (Schritt a9). Außerdem zeigt die Testaufgabeneinrichtung im ersten Anzeigebereich ein Graphiksymbol an, das einen Testbedingungsdateinamen darstellt, den der ausgewählte Testgegenstand zuordnet (Schritt a10).

Anschließend ermittelt die Testaufgabeneinrichtung, ob der Benutzer des Testsystems ein Graphiksymbol des Testbedingungsdateinamens auswählt (Schritt a11). Wenn dieses Graphiksymbol ausgewählt wird, bewegt die Testaufgabeneinrichtung das Graphiksymbol, das den ausgewählten Testbedingungsdateinamen darstellt, zu einer Position im ersten Anzeigebereich, an der das ausgewählte Graphiksymbol angezeigt wird (Schritt a12), und setzt den ausgewählten Testbedingungsdateinamen im Testziel, den die Testaufgabe besitzt, zum Zuordnen (Schritt a13).

Wenn im Schritt a7 oder allen obenbeschriebenen Schritten die Ermittlung negativ ist, ermittelt die Testaufgabeneinrichtung alternativ, ob der Benutzer des Testsystems den Hochgeschwindigkeitstest auswählt (Schritt a14), und verbindet dann, wenn dies zutrifft, das Graphiksymbol, das die DUT darstellt, mit einem Graphiksymbol, das einen Testelementnamen darstellt, der dem Hochgeschwindigkeitstest zugeordnet ist. Wenn das Graphiksymbol, das die DUT darstellt, mit dem Graphiksymbol verbunden ist, das den Testelementnamen darstellt, der dem Hochgeschwindigkeitstest zugeordnet ist, erzeugt die Testaufgabe den Testkompromiß, der die Kompromißinformationen des Hochge-

schwindigkeitstests im Speicher als Attribut speichert, und setzt das Testziel, das es zuordnet, in die Zuordnungsbeziehung (Schritt a15). Anschließend sendet die Testaufgaben-einrichtung das erzeugte Testziel zur virtuellen Instrumenteneinrichtung, um den Test zu starten.

Wenn andererseits der Benutzer des Testsystems den hochgenauen Test auswählt, verbindet die Testaufgabeneinrichtung das Graphiksymbol, das die DUT darstellt, mit dem Graphiksymbol, das den Testelementnamen darstellt, der dem hochgenauen Test zugeordnet ist. Wenn das Graphiksymbol, das die DUT darstellt, mit dem Graphiksymbol, das den Testelementnamen darstellt, der dem hochgenauen Test zugeordnet ist, verbunden ist, erzeugt die Testaufgabe den Testkompromiß, der die Kompromißinformationen des hochgenauen Tests im Speicher als Attribut speichert und veranlaßt, daß das Testziel den Testkompromiß zuordnet (Schritt a16). Anschließend sendet die Testaufgabeneinrichtung das erzeugte Testziel zur virtuellen Instrumenteneinrichtung, um den Test zu starten.

Die Serie der obenbeschriebenen Prozesse dient zum Definieren des Zwecks des Tests.

#### [Testparameterspeichereinrichtung]

Die Testparameterspeichereinrichtung speichert im Speicher Testparameter, die zur Durchführung des Tests mit dem Testelement erforderlich sind. Die Fig. 8 ist ein Schaubild, das eine Konfiguration eines Basisobjekts zeigt, das zur Implementierung der Testparameterspeichereinrichtung erforderlich ist. Wie in Fig. 8 gezeigt, umfassen die Basisobjekte, die die Testparameterspeichereinrichtung implementieren, einen "Testparametergegenstand" und einen "Testparameter". Im folgenden wird jedes Objekt beschrieben.

#### Testparametergegenstand

Dies ist ein Objekt zum Speichern eines Testparameternamens und eines "Testparameters" im Speicher als Attribute.

#### Testparameter

Dieses ordnet wenigstens einen oder mehrere Testparametergegenstände zu, die zur Durchführung des Tests mit dem Testelement erforderlich sind. Es enthält ferner Testdaten zum Speichern der durch Verwendung des Testparameters erhaltenen Daten.

#### [Testdatenspeichereinrichtung]

Diese speichert im Speicher Daten, die das virtuelle Instrument im Test aufnimmt. Wie in Fig. 8 gezeigt, sind Testdaten ein Basisobjekt, das die Testdatenspeichereinrichtung implementiert.

#### Testdaten

Diese speichern im Speicher die Testdaten, die das virtuelle Instrument im Test aufnimmt, als Attribut. Die Testdaten halten einen Testparameter, der verwendet wird, um die Testdaten aufzunehmen.

#### [Virtuelle Instrumenteneinrichtung]

Die virtuelle Instrumenteneinrichtung dient zum Durchführen eines Tests für einen Zweck/Ziel. Die virtuelle Instrumenteneinrichtung erzeugt einen Testparameter aus einer Meßzielinformation, die durch eine Testaufgabeneinrichtung spezifiziert wird, indem sie eine Testklassenein-

richtung verwendet, und sendet den erzeugten Testparameter zu einem Testelement, um das Testelement zu steuern und den Test durchzuführen. Anschließend nimmt sie die Testdaten vom Testelement auf und speichert die aufgenommenen Daten mittels der Testdatenspeichereinrichtung im Speicher. Wie in Fig. 8 gezeigt, ist ein Basisobjekt, das die virtuelle Instrumenteneinrichtung implementiert, ein "virtuelles Instrument".

#### Virtuelles Instrument

Das virtuelle Instrument ordnet mehrere Testdaten und Testparameter zu. Außerdem ordnet es mehrere "Darstellungsblätter" zu zum Anzeigen des Testergebnisses auf der graphischen Anzeigevorrichtung. Ferner ordnet es mehrere "virtuelle Instrumentendienste" zu zum Ausführen von Diensten wie z. B. einer arithmetischen Operation mit den Testdaten. Das virtuelle Instrument verarbeitet das Starten der mit einem "Meßziel" spezifizierten Messung.

#### [Darstellungsblatteinrichtung]

Die Darstellungsblatteinrichtung ist eine Einrichtung zum Darstellen von Daten, die vom Testelement aufgenommen worden sind. Außerdem kann sie eine "Schlitzinformation" von einer Bereichsspezifikation auf der graphischen Darstellung mittels der Zeigevorrichtung erzeugen, indem sie die Testdatenspeichereinrichtung und die Schlitzinformationseinrichtung kombiniert. Wie in Fig. 8 gezeigt, ist ein Basisobjekt zur Implementierung der Darstellungsblatteinrichtung ein Darstellungsblatt.

#### Darstellungsblatt

Das Darstellungsblatt ordnet Testdaten und das virtuelle Instrument zu, das die Testdaten aufnimmt, und stellt den Inhalt der Testdaten in einem Graphen dar. Außerdem kann das Darstellungsblatt die Schlitzinformationen von einer Bereichsspezifikation auf der graphischen Darstellung mittels der Zeigevorrichtung erzeugen. Das Darstellungsblatt ordnet mehrere Schlitzinformationen zu. Das Darstellungsblatt besitzt eine Funktion zum Unterstützen einer Operation auf der graphischen Darstellung, die auf ein Graphiksymbol des virtuellen Instrumentendienstes zugreift, der das virtuelle Instrument zuordnet, oder auf eine "Wechselwirkungsvorrichtung", um die verfügbaren Testelemente anzuzeigen.

#### [Schlitzinformationseinrichtung]

Die Schlitzinformationseinrichtung ist ein Objekt zum Speichern eines auf der graphischen Darstellung mittels der Zeigevorrichtung spezifizierten Bereiches im Speicher. Wie in Fig. 8 gezeigt, ist ein Basisobjekt zur Implementierung der Schlitzinformationseinrichtung die Schlitzinformation.

#### Schlitzinformation

Die Schlitzinformation ist ein Objekt zum Speichern von Bereichsdaten im Speicher als Attribut. Sie ordnet Testdaten zu, die eine Quelle der Bereichsdaten ist. Außerdem besitzt sie eine Funktion zum Anzeigen eines äußeren Rahmens von Bereichsdaten auf der graphischen Darstellung der zugeordneten Testdaten. Ferner besitzt die Schlitzinformation eine Funktion zum Speichern eines virtuellen Instrumentendienstes, der in einem Bereich ausgewählt worden ist, der durch die Bereichsdaten auf der graphischen Darstellung angegeben wird, und eines virtuellen Instruments im Speicher als Attribut.

Die virtuelle Instrumentendiensteinrichtung ist eine Einrichtung, die Dienste zur Verfügung stellt, wie z. B. eine arithmetische Operation für einen Bereich, der auf der graphischen Darstellung des Testergebnisses spezifiziert wird. Die Dienste, wie z. B. die arithmetische Operation auf einem Bereich der graphischen Darstellung, werden erhalten durch Kombinieren der virtuellen Instrumenteneinrichtung, der Schlitzinformationseinrichtung, der Testklasseneinrichtung und der Testdaten Speichereinrichtung. Wie in Fig. 8 gezeigt, ist ein Basisobjekt zur Implementierung der virtuellen Instrumentendiensteinrichtung ein "virtueller Instrumentendienst".

#### Virtueller Instrumentendienst

Der virtuelle Instrumentendienst ist ein Objekt, das einen Dienstprozeß wie z. B. eine arithmetische Operation auf dem spezifizierten Bereich der Testdaten, die vom virtuellen Instrument aufgenommen worden sind, zur Verfügung stellt. Er besitzt einen "Dienstprozeß".

Der virtuelle Instrumentendienst enthält z. B. eine "Spitzenwertsuche", um einen Spitzenwert innerhalb eines spezifizierten Bereichs der Testdaten zu finden, sowie eine "Leistungsberechnung", zum Auffinden der Leistung innerhalb eines spezifizierten Bereichs der Testdaten. Die Spitzenwertsuche und die Berechnung der Leistung sind Objekte, die vom virtuellen Instrumentendienst abgeleitet sind. Der Dienstprozeß des virtuellen Instrumentendienstes ist ein Prozeß, der für Abkömmling-Objekte wie z. B. die Spitzenwertsuche und die Berechnung der Leistung definiert ist. Unabhängig davon, ob ein verwendetes Objekt der Dienstprozeß oder die Spitzenwertsuche ist, wird es als der virtuelle Instrumentendienst behandelt. Wenn der Dienstprozeß ausgeführt wird, wird der Dienstprozeß für den Dienstprozeß ausgeführt, oder der Dienstprozeß wird für die Spitzenwertsuche ausgeführt, so daß ein Ergebnis der Operation des Dienstprozesses oder der Spitzenwertsuche erhalten werden kann.

#### [Wechselwirkungseinrichtung]

Die Wechselwirkungseinrichtung ist eine Einrichtung zum Durchführen einer Anfrage für einen neuen Testgegenstand, zum Starten eines neuen Tests und einer Graphikdarstellung der neuen Testdaten durch Kombinieren der virtuellen Instrumenteneinrichtung, der Schlitzinformationseinrichtung und der Testklasseneinrichtung. Sie besitzt zusätzlich eine Funktion zum Anzeigen eines Graphiksymbols, das das Testelement darstellt, das im Testsystem verfügbar ist. Wie in Fig. 8 gezeigt, ist ein Basisobjekt, das die Wechselwirkungseinrichtung implementiert, eine Wechselwirkungsvorrichtung.

#### Wechselwirkungsvorrichtung

Sie verwendet ein virtuelles Instrument, das einem spezifizierten Testelementnamen zugeordnet ist, und eine "Testklasse", die einem spezifizierten Testgegenstand zugeordnet ist, um einen Testparameter zu erzeugen, und sendet den Testparameter zum Testen zum virtuellen Instrument. Die Wechselwirkungsvorrichtung ist ein Objekt, das mehrere verfügbare virtuelle Instrumente zuordnet.

#### [Testklasseneinrichtung]

Die Testklasseneinrichtung ist eine Einrichtung zum Um-

setzen eines Zwecks/Ziels in einen Testparameter des Testelements. Außerdem besitzt sie eine Funktion zum Berechnen eines Testergebnisses entsprechend dem Testgegenstand aus den Daten, die vom Testelement aufgenommen wurden, unter Verwendung des umgesetzten Testparameters. Die Testklasseneinrichtung antwortet auf eine Anforderung von der virtuellen Instrumenteneinrichtung oder der Wechselwirkungseinrichtung und erzeugt Testparameter für ein spezifiziertes Testelement aus den Informationen, wie z. B. einer Testbedingung, dem Testkompromiß, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt wird, oder einem Testbereich, der den X- und Y-Achsen entspricht. Außerdem berechnet sie ein Testergebnis entsprechend einem Testgegenstand aus den Testdaten als Antwort auf eine Anforderung von der virtuellen Instrumenteneinrichtung.

Die Fig. 9 ist ein Flußdiagramm, das eine Konfiguration eines Basisobjekts zeigt, das zur Implementierung der Testklasseneinrichtung erforderlich ist. Wie in der Figur gezeigt, enthalten die Basisobjekte zur Implementierung der Testklasseneinrichtung eine Testklasse, eine "Blatttestklasse", einen "Testklassenimplementierer" und eine "zusammengesetzte Testklasse".

#### Testklasse

Diese ist ein abstraktes Objekt, das einen Prozeß definiert, der ein Abkömmling-Objekt der Testklasse liefert. Dieses Objekt definiert Prozesse, die ein Abkömmling-Objekt zur Verfügung stellt, wie z. B. der Erzeugung von Testparametern für ein spezifiziertes Testelement, auf der Testbedingung und Informationen, wie z. B. einem Testbereich, der den X- und Y-Achsen zugeordnet ist, sowie die Berechnung des Testergebnisses entsprechend dem Meßgegenstand aus den Testdaten.

#### Blatttestklasse

Dies ist ein Objekt zum Erzeugen eines abstrakten Testparameters, wenn der Testgegenstand einen Basisrest des Testelements entspricht. Die Blatttestklasse ordnet ferner einen Testklassenimplementierer zu zum Umsetzen des abstrakten Testparameters in einen Testparameter des virtuellen Instruments. Wenn z. B. das Testelement ein Spektrumanalysierer ist, ist die Blatttestklasse die "Leistungsspektrummessung" zum Messen des Spektrums in einem Frequenzband, oder die "Linienspektrummessung" zum Messen des Spektrums bei einer Frequenz. Eine solche Leistungsspektrummessung oder Linienspektrummessung ist ein Objekt, das von der Blatttestklasse abgeleitet ist.

#### Testklassenimplementierer

Die Leistungsspektrummessung kann nicht nur mittels des Spektrumanalysierers gemessen werden, sondern auch von mehreren Testelementen wie z. B. einem FFT-Analysierer. Diese Testelemente besitzen unterschiedliche Testparameter. Zum Beispiel ist ein Spektrumanalysierer eine Frequenzauflösungsbandbreite ein Testparametergegenstand zum Ermitteln der Messungsauflösung, während ein FFT-Analysierer die Anzahl der Anzeigepunkte ein Testparametergegenstand zum Ermitteln der Messungsauflösung ist. Es wird somit notwendig, ein Objekt zu besitzen zum Umsetzen eines abstrakten Testparameters, der von der Blatttestklasse erzeugt worden ist, in einen spezifischen Testparameter für jedes virtuelle Instrument. Der Testklassenimplementierer ist ein Objekt zum Umsetzen desselben in einen spezifischen Testparameter für jedes virtuelle Instrument unter



Verwendung des virtuellen Instruments. Er ist ein Objekt, das vom Testklassenimplementierer abgeleitet ist, der für die Umsetzung in einen spezifischen Testparameter des virtuellen Instruments verantwortlich ist.

#### Zusammengesetzte Testklasse

Dies ist ein Objekt zum Erzeugen eines abstrakten Testparameters, wenn der Testgegenstand eine Kombination von Basistests des Testelements ist. Zum Beispiel dient die Oberwellenmessung zum Messen der Leistungsdifferenz zwischen einem Basissignal und den Oberwellen. Die Oberwellenmessung kann die Leistung jedes Signals mit einem Durchlauf des Testelements messen, wenn in einem Frequenzband gemessen wird, das das Basissignal und die Oberwellen enthält. Außerdem kann die Messung mit hoher Genauigkeit durchgeführt werden, indem das Basissignal und die Oberwellensignale mit einem Testelement getrennt durchlaufen werden. Die erste Messung kann von der Leistungsspektrummessung gemessen werden, die von der Blatestestklasse abgeleitet worden ist. Die letztere Messung kann von der Linienspektrummessung gemessen werden, die von der Blatestestklasse abgeleitet worden ist. Zum Beispiel wird ein Test zum Testen des Basissignals und des Oberwellensignals mit hoher Geschwindigkeit mit einem Durchlauf als Hochgeschwindigkeitstest bezeichnet, während ein Test zum Testen des Basissignals und des Oberwellensignals in getrennten Durchläufen mit hoher Genauigkeit als hochgenauer Test bezeichnet wird. Die zusammengesetzte Testklasse ist ein Objekt, das mehrere Blatestestklassen für einen solchen Hochgeschwindigkeitstest und einen hochgenauen Test zuordnet. Außerdem ist die Oberwellenmessung ein Objekt, das von der zusammengesetzten Testklasse abgeleitet ist.

#### [Testplaneinrichtung]

Die Testplaneinrichtung erzeugt ein Programm oder arbeitet dieses ab, das die Ausführungsreihenfolge der Testaufgaben steuert durch Kombinieren von Programmelementen und zusammengesetzten Programmenelementen. Die Testaufgabe ist eines der Programmenelemente. Die Testplaneinrichtung zeigt diese Programmenelemente und die zusammengesetzten Programmenelemente auf der graphischen Anzeigevorrichtung an, wodurch eine GUI geschaffen wird, die dem Benutzer des Testsystems erlaubt, ein Programm zu erzeugen oder laufen zu lassen. Die Fig. 10 ist ein Schaubild, das eine Konfiguration eines Basisobjekts zeigt, das zur Implementierung der Testplaneinrichtung erforderlich ist. Wie in der Figur gezeigt, enthalten die Basisobjekte, die die Testplaneinrichtung implementieren, die Objekte "Aufgabenkomponente", "Beginn", "Ende", "Pause", "Warten", "warten bis zum spezifizierten Zeitpunkt", "zusammengesetzte Aufgabe", "Schleife", "Schalter", "bedingte Schleife" und "Testplan".

#### Aufgabenkomponente

Dies ist ein Objekt zum abstrakten Behandeln eines Programmenelements (alle Programmenelemente und zusammengesetzten Elemente, die von der Aufgabenkomponente abgeleitete Objekte sind). Die Aufgabenkomponente besitzt einen abstrakten Ausführungs-Prozess. Es ist ein Objekt des Programmenelements, das von der Aufgabenkomponente abgeleitet ist, welches den spezifischen Ausführungs-Prozess implementiert. Wie oben in der "Vererbung des Objekts" beschrieben, kann dann, wenn ein von der Aufgabenkomponente abgeleitetes Programmenelement als eine Aufgaben-

komponente behandelt wird, der Ausführungs-Prozess ausgeführt werden, unabhängig davon, was das Programmenelement ist.

#### Beginn

Dies ist ein Programmenelement, das den Beginn eines Programms anzeigt. Es ist ein Programmenelement mit einem Ausführungs-Prozess zum Erfassen eines Testplans, der den Beginn zuordnet, und zum Starten des Ausführungs-Prozesses des erfaßten Testplans.

#### Ende

Dies ist ein Programmenelement, das das Ende eines Programms anzeigt. Der Ausführungsprozeß von Ende bewirkt nichts.

#### Pause

Dies ist ein Programmenelement mit einem Ausführungs-Prozess, der ein Programm anhält, bis eine Eingabe von der Zeigevorrichtung vorliegt.

#### Warten

Dies ist ein Programmenelement mit einem Ausführungs-Prozess, der ein Programm für eine spezifizierte Zeitperiode anhält. Das Warten speichert eine Programmanzahlzeit-spanne im Speicher als Attribut.

#### Warten bis zum spezifizierten Zeitpunkt

Dies ist ein Programmenelement mit einem Ausführungs-Prozess, der ein Programm bis zu einem spezifizierten Zeitpunkt anhält. Das "Warten bis zu einem spezifizierten Zeitpunkt" wird im Speicher als Attribut gespeichert.

#### Zusammengesetzte Aufgabe

Dies ist ein Objekt zum abstrakten Behandeln eines zusammengesetzten Programmenelements. Die zusammengesetzte Aufgabe ist ein Objekt, das mehrere Aufgabenkomponenten zuordnet. Die Zuordnungsbeziehung speichert einen Einsprungpunkt jedes Objektes im Speicher als Matrixdaten. Der Ausführungs-Prozess der zusammengesetzten Aufgabe führt die Ausführungs-Prozesse der Aufgabenkomponenten, die gehalten werden, in einer Reihenfolge beginnend von der Spitze der Matrixdaten aus.

#### Schleife

Dies ist ein zusammengesetztes Programmenelement mit einem Ausführungs-Prozess, der den Ausführungs-Prozess der zusammengesetzten Aufgabe in einer spezifizierten Häufigkeit ausführt. Die Schleife speichert die Anzahl der Ausführungen im Speicher als Attribut.

#### Schalter

Dies ist ein zusammengesetztes Programmenelement mit einem Ausführungs-Prozess, der den Ausführungs-Prozess der enthaltenen zusammengesetzten Aufgabe startet, wenn eine Bedingung erfüllt ist. Der Schalter speichert eine Bedingung im Speicher als Attribut.

Dies ist ein zusammengesetztes Programmelement mit einem Ausführungs-Prozß der enthaltenen zusammengesetzten Aufgabe, bis eine spezifizierte Bedingung erreicht ist. Die bedingte Schleife speichert eine Bedingung im Speicher als Attribut.

### Testplan

Der Testplan ist ein Programm, das die Programmelemente vom Beginn bis zum Ende unter Verwendung der zusammengesetzten Aufgabe zuordnet. Ein Ausführungs-Prozß des Testplans führt den Ausführungs-Prozß der enthaltenen zusammengesetzten Aufgabe aus. Der Testplan ordnet mehrere Aufgabenkomponenten zu, um verfügbare Programmelemente zu speichern.

Außerdem ist eine Testaufgabe ferner ein Programmelement, das von der Aufgabenkomponente abgeleitet ist. Die Programmierung eines Testsystems gemäß der vorliegenden Erfindung besteht im Erstellen eines Testplans. Es ist nur die Testaufgabe, die ein Programmelement ist, das einen Test durchführt. Die anderen Programmelemente außer der Testaufgabe dienen zum Steuern der Reihenfolge der Ausführung der Testaufgaben oder dergleichen.

Die Testplaneinrichtung verwendet den Testplan, um dem Benutzer des Testsystems die GUI zur Verfügung zu stellen, um eine solche Programmierung durchzuführen. Die Fig. 11 ist ein Flußdiagramm, das die Schritte eines Prozesses zeigt, der von der Testplaneinrichtung ausgeführt wird. Die Testplaneinrichtung erzeugt einen zweiten Anzeigebereich, der auf der graphischen Anzeigevorrichtung überlappend angezeigt werden kann. Hierbei wird angenommen, daß der erste Anzeigebereich von der Testaufgabeneinrichtung erzeugt wird, wobei im ersten Anzeigebereich bereits ein Graphiksymbol der Meßaufgabe angezeigt wird.

Zuerst zeigt die Testplaneinrichtung ein Graphiksymbol des "verfügbaren Programmelements", das vom Testplan gehalten wird, auf dem zweiten Anzeigebereich an (Schritt b1), zeigt Graphiksymbole von Beginn und Ende, die vom Testplan gehalten werden, auf dem zweiten Anzeigebereich an (Schritt b2) und verbindet die Anzeigen der Graphiksymbole von Beginn und Ende auf dem zweiten Anzeigebereich (Schritt b3).

Anschließend wählt die Testplaneinrichtung eines der Testaufgaben-Graphiksymbole auf dem ersten Anzeigebereich, der von der Testaufgabeneinrichtung erzeugt wird, oder ein Programmelement-Graphiksymbol, das auf dem zweiten Anzeigebereich angezeigt wird (Schritt b4).

Anschließend wählt sie eine Position, an der das ausgewählte Programmelement-Graphiksymbol erzeugt wird und zwischen den Graphiksymbolen Beginn und Ende angezeigt wird (Schritt b5).

Anschließend löscht die Testplaneinrichtung alle Verbindungsanzeigen zwischen den Graphiksymbolen Beginn und Ende (Schritt b6) und erzeugt das Graphiksymbol des ausgewählten Programmelements an einer spezifizierten Position im zweiten Anzeigebereich. Anschließend richtet sie die Graphiksymbole von Beginn zum Ende aus (Schritt b8) und verbindet zwischen benachbarten Graphiksymbolen von Beginn zum Ende und zeigt dies im zweiten Anzeigebereich an (Schritt b9).

Anschließend speichert sie das Programmelement in der Zuordnungsbeziehung der zusammengesetzten Aufgabe, die im Testplan enthalten ist (Schritt b10). In diesem Fall ordnet sie die Matrixdaten der Einsprungpunkte der Programmelemente neu an, die als die Zuordnungsbeziehung gespeichert werden sollen, so daß die Programmelemente

zwischen den Graphiksymbolen Beginn und Ende mit der Reihenfolge der Anzeige der Graphiksymbole übereinstimmen. Wenn ein neues Programmelement ausgewählt wird, kehrt der Prozeß zur Wiederholung zum Schritt b4 zurück. Mit einer solchen Operation kann ein Programm erstellt werden.

Im folgenden werden Funktionen und Vorteile beschrieben, die durch Kombinieren dieser obenbeschriebenen Einrichtungen erreicht werden können.

- (1) Kombination der Testaufgabeneinrichtung, der Testparameterspeichereinrichtung, der Testdatenspeichereinrichtung, der virtuellen Instrumenteneinrichtung, der Darstellungsblatteinrichtung und der Testklasseneinrichtung (erste Kombination)

Die Testaufgabeneinrichtung erlaubt, eine Operation ausgehend von einem Test-Zweck/Ziel durchzuführen. Ein virtuelles Instrument wird für ein zu testendes Element als Ergebnis der Operation ausgewählt, wodurch ein Testziel, das dem Test-Zweck/Ziel entspricht, erzeugt wird. Die Testaufgabeneinrichtung leitet das erzeugte Testziel zum ausgewählten virtuellen Instrument der virtuellen Instrumenteneinrichtung weiter, um den Test zu starten.

Die Anzeige des Tests und des Testergebnisses wird durchgeführt von der Struktur, die eine Kombination aus virtueller Instrumenteneinrichtung, Testparameterspeichereinrichtung, Testklasseneinrichtung, Testdatenspeichereinrichtung und Darstellungsblatteinrichtung ist. Zuerst verwendet das virtuelle Instrument eine Testklasse, die einem Testgegenstand entspricht, die das von der Testaufgabeneinrichtung übergebene Testziel zuordnet, um einen Testparameter zu erzeugen. Anschließend erzeugt das virtuelle Instrument Testdaten und ein Darstellungsblatt zum Anzeigen der Testdaten. Danach setzt das virtuelle Instrument den Testparameter in das Testelement ein, führt den Test durch, indem es das Testelement steuert, und erhält die Testdaten vom Testelement. Anschließend speichert das virtuelle Instrument die vom Testelement erhaltenen Testdaten in den Testdaten und fordert das Darstellungsblatt an, um die Testdaten anzuzeigen. Schließlich verwendet das virtuelle Instrument die Testklasse, um ein Testergebnis entsprechend dem Testgegenstand aus den vom Testelement erhaltenen Testdaten zu berechnen.

Wie oben beschrieben ist, kann die Operation ausgehend vom Test-Zweck/Ziel durchgeführt werden, indem die Testaufgabeneinrichtung, die Testparameterspeichereinrichtung, die Testdatenspeichereinrichtung, die virtuelle Instrumenteneinrichtung, die Darstellungsblatteinrichtung und die Testklasseneinrichtung kombiniert werden, wodurch ein Testsystem erhalten werden kann, das einen Test auf der Grundlage eines Test-Zwecks/Ziels durchführt und ein Testergebnis anzeigt (erstes Objekt). Außerdem kann die Testaufgabeneinrichtung eine Funktion besitzen, die als Test-Zweck/Ziel den Kompromiß spezifiziert, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt werden soll (zweites Objekt).

Wie oben beschrieben worden ist, verwendet ein Programmierverfahren eines herkömmlichen Testsystems einen Lösungsansatz zum Verbinden von Programmabschnitten, die die Datenaufnahme, die arithmetische Operation oder die graphische Darstellung durchführen, durch Zeigen auf Punkte, die für entsprechende Graphiksymbole vorgegeben sind. Ein solcher Verbindungslösungsansatz erfordert ein vollständiges Verständnis der Funktionen der Graphiksymbole und einer Funktion jedes Verbindungspunktes, so daß dieses Programmierverfahren nicht leicht von jedermann verwendet werden kann. Außerdem kann ein solches Ver-

such-Irrtum-Verfahren nicht durchgeführt werden, sofern nicht der Benutzer des Testsystems die Hardware-Konfiguration der Testvorrichtung und die entsprechenden Testparameter vollständig verstanden hat. Wenn andererseits die obenbeschriebenen Einrichtungen kombiniert werden, kann ein Testsystem geschaffen werden, das die Testklasseneinrichtung benutzt, die einen Testparameter für ein Testelement erzeugt durch Auswählen eines Testgegenstandes, einer Testbedingung und des Testkompromisses, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt werden soll, was mit dem herkömmlichen Testsystem nicht erreicht werden konnte, wodurch die Operation ausgehend von einem Test-Zweck/Ziel, das den Testkompromiß enthält, durchgeführt werden kann, der Test auf der Grundlage des Test-Zwecks/Ziels ausgeführt werden kann und das Testergebnis angezeigt werden kann.

Folglich kann der Benutzer des Testsystems den Test ohne Kenntnis der Hardware-Konfiguration der Testausrüstung oder von Einzelheiten der entsprechenden Testparameter und Programmabschnitte des Testsystems durchführen, so daß er sich auf die Einzelheiten der Bewertung der DUT konzentrieren kann.

(2) Kombination aus Testaufgabeneinrichtung, Testparameter-einrichtung, Testdaten Speichereinrichtung, virtueller Instrumenteneinrichtung, Darstellungsblatteinrichtung und Testklasseneinrichtung, sowie Schlitzinformationseinrichtung und virtueller Instrumenteneinrichtung (zweite Kombination)

Dies ist eine Kombination aus Schlitzinformationseinrichtung und virtueller Instrumenteneinrichtung und der obenbeschriebenen ersten Kombination. Die Schlitzinformationseinrichtung ermöglicht zusammen mit der Darstellungsblatteinrichtung, daß der Benutzer einen Bereich auf einer graphischen Darstellung mittels einer Zeigevorrichtung auswählt. Außerdem bietet die virtuelle Instrumenteneinrichtung Dienste an, wie z. B. die Berechnung innerhalb eines Bereiches, der auf der graphischen Darstellung des Testergebnisses spezifiziert ist, auf der Grundlage der Schlitzinformation. Dementsprechend ist es möglich, eine Funktion zu erhalten, die dann, wenn Koordinaten auf einer graphischen Darstellung des Testergebnisses spezifiziert werden, einen Dienst starten kann, der auf den spezifizierten Bereich gerichtet ist. Diese Dienste werden durch eine Operation auf dem ersten Anzeigebereich gestartet, die von der Testaufgabeneinrichtung auf der graphischen Anzeigevorrichtung erzeugt wird. Im folgenden werden die Schritte zum Starten eines Dienstes beschrieben, der auf den spezifizierten Bereich gerichtet ist.

Zuerst erzeugt das Darstellungsblatt dann, wenn auf der graphischen Darstellung mittels der Zeigevorrichtung ein Bereich ausgewählt wird, die Schlitzinformation zum Speichern des spezifizierten Bereiches im Speicher. Anschließend überlappt die Schlitzinformation einen äußeren Rahmen eines ausgewählten Bereiches auf der graphischen Darstellung und zeigt diesen an.

Anschließend greift das Darstellungsblatt auf ein virtuelles Instrument zu, das es zuordnet. Es zeigt anschließend im ersten Anzeigebereich ein Graphiksymbol des virtuellen Instrumentendienstes an, den das virtuelle Instrument zuordnet. Sobald der virtuelle Instrumentendienst ausgewählt ist, zeichnet die Schlitzinformation den ausgewählten virtuellen Instrumentendienst im Attribut der Schlitzinformation auf. Die Schlitzinformation zeigt das ausgewählte Graphiksymbol auf der graphischen Darstellung an und führt den Dienstprozeß aus, den der virtuelle Instrumentendienst besitzt.

Der virtuelle Instrumentendienst führt einen Dienst aus,

wie z. B. die Berechnung des ausgewählten Bereiches der graphischen Darstellung, wenn der Dienstprozeß ausgeführt wird, und zeigt ein Ergebnis des Dienstes im ersten Anzeigebereich an. Das heißt, die zweite Kombination kann das dritte Objekt bilden.

Wenn jedoch z. B. beabsichtigt ist, einen Leistungswert der Störung mit dem herkömmlichen Testsystem zu erhalten, ist es erforderlich, eine Programmierung und eine Operation erneut durchzuführen, so daß a) der Test beendet wird, b) ein Programm zum Hinzufügen von Testdaten erstellt wird, und c) der Test erneut durchgeführt wird, um den Leistungswert zu finden. Wenn in diesem Fall die Störung, von der die Leistung bestimmt wird, ein Teil der Testdaten ist, ist es erforderlich, Matrixdaten, die den Positionskorrespondenzen der Störung auf der graphischen Darstellung zu lokalisieren und ein Programm hinzuzufügen, das die lokalisierten Matrixdaten hinzufügt. Somit erfordert das herkömmliche Testsystem sehr viel mehr Arbeitsaufwand, selbst beim lediglichen Auffinden des Leistungswertes der Störung auf der graphischen Darstellung.

Andererseits kann gemäß der zweiten Kombination der Dienst, wie z. B. die Berechnung des ausgewählten Bereiches auf der graphischen Darstellung, nur durch Spezifizieren von Koordinaten auf der graphischen Darstellung des Testergebnisses unter Verwendung einer Zeigevorrichtung und durch Auswählen des Graphiksymbols des virtuellen Instrumentendienstes durchgeführt werden, wodurch das vom Dienst erhaltene Ergebnis auf der Graphikanzeigevorrichtung angezeigt wird.

Ein Leistungswert der Störung, der ein Teil der Testdaten ist, kann vom herkömmlichen Testsystem nicht gefunden werden, sofern nicht ein obenbeschriebenes Programm erstellt wird. Die Verwendung der zweiten Kombination ermöglicht jedoch, eine solche Operation mit nur zwei Operationen der Zeigevorrichtung ohne Modifizierung des Programms insgesamt durchzuführen.

(3) Kombination aus Testaufgabeneinrichtung, Testparameter Speichereinrichtung, Testdaten Speichereinrichtung, virtueller Instrumenteneinrichtung, Darstellungsblatteinrichtung und Testklasseneinrichtung, sowie Schlitzinformationseinrichtung und Wechselwirkungseinrichtung (dritte Kombination)

Dies ist eine Kombination aus der Schlitzinformationseinrichtung und der Wechselwirkungseinrichtung mit der obenbeschriebenen ersten Kombination. Die Schlitzinformationseinrichtung ermöglicht zusammen mit der Darstellungsblatteinrichtung, daß der Benutzer einen Bereich auf der graphischen Darstellung mittels einer Zeigevorrichtung auswählen kann. Außerdem führt die Wechselwirkungseinrichtung eine Anfrage für einen neuen Testgegenstand, den Start eines neuen Testes und eine graphische Darstellung der neuen Testdaten durch mittels Kombinieren der virtuellen Instrumenteneinrichtung, der Schlitzinformationseinrichtung und der Testklasseneinrichtung. Es ist daher möglich, a) eine Funktion zum Erzeugen eines neuen Testparameters für ein Testelement aus einem spezifizierten Bereich auf der graphischen Darstellung des vom Testelement erhaltenen Ergebnisses ohne Umprogrammierung eines Programms und zum Starten eines neuen Testes mit dem erzeugten Testparameter zu erhalten, sowie b) eine Funktion zu erhalten zum Erzeugen eines neuen Testparameters für ein Testelement aus einem spezifizierten Bereich auf der graphischen Darstellung des von einem weiteren Testelement erhaltenen Testergebnisses und zum Starten eines neuen Testes mit dem erzeugten Testparameter. Im folgenden werden die Schritte

zum Erhalten der Funktionen a und b beschrieben.

Zuerst erzeugt das Darstellungsblatt dann, wenn ein Bereich auf der graphischen Darstellung mittels der Zeigevorrichtung ausgewählt wird, eine Schlitzinformation zum Speichern des spezifizierten Bereichs im Speicher. Anschließend überlappt die Schlitzinformation einen äußeren Rahmen des ausgewählten Bereiches auf der graphischen Darstellung und zeigt diesen an.

Anschließend greift das Darstellungsblatt auf die Wechselwirkungsrichtung zu. Es zeigt anschließend im ersten Anzeigebereich ein Graphiksymbol eines Testelementnamens an, der dem virtuellen Instrumentendienst zugeordnet ist, den die Wechselwirkungsrichtung zuordnet. Sobald das virtuelle Instrument ausgewählt ist, zeichnet die Schlitzinformation den ausgewählten Testelementnamen im Attribut der Schlitzinformation auf. Die Schlitzinformation zeigt das ausgewählte Graphiksymbol auf der graphischen Darstellung an und fordert die Wechselwirkungsrichtung zu einer Messung innerhalb eines spezifizierten Bereiches auf.

Die Wechselwirkungsrichtung greift auf den ausgewählten Testelementnamen zu, und startet dann, wenn ein Testgegenstand, der als Vorgabe gestartet wird, im Attribut des Testelementnamens gespeichert ist, die Messung mit dem gespeicherten Testgegenstand. Wenn im Attribut des Testelementnamens kein Testgegenstand gespeichert ist, der als Vorgabe gestartet wird, greift die Wechselwirkungsrichtung dann, wenn auf den ausgewählten Testelementnamen zugegriffen wird, auf einen spezifizierten Testelementnamen zu und zeigt ein Graphiksymbol des Testgegenstandes an, den der Testelementnamen zuordnet. Anschließend startet die Wechselwirkungsrichtung dann, wenn das Graphiksymbol des Testgegenstandes ausgewählt ist, die Messung mit dem ausgewählten Testgegenstand.

Wenn die Wechselwirkungsrichtung die Messung startet, sendet sie zuerst die Schlitzinformation zu einer Meßklasse, die dem Testgegenstand zugeordnet ist, und erzeugt einen Testparameter eines virtuellen Instruments, das dem ausgewählten Testelementnamen zugeordnet ist. Anschließend sendet die Wechselwirkungsrichtung den erzeugten Parameter zu einem virtuellen Instrument, das dem ausgewählten Testelementnamen zugeordnet ist, und empfängt die Testdaten als Ergebnis des durchgeführten Tests vom virtuellen Instrument. Schließlich stellt die Wechselwirkungsrichtung den Inhalt der Testdaten unter Verwendung der Darstellungsblatteinrichtung als Graphen dar. Somit ermöglicht die Verwendung der dritten Kombination, die vierten und fünften Objekte zu erhalten.

Wenn außerdem Frequenzen unerwarteter Störungen z. B. genauer analysiert werden, kann das herkömmliche Testsystem eine Umgebung erhalten, die das Zurücksetzen der Testparameter durch Programmierung erlaubt, um eine Instrumententafel anzuzeigen, und um die Hardware für die Mittenfrequenz oder die Frequenzspanne zu steuern. Um jedoch mehrere unerwartete Störungen zu analysieren, muß der Benutzer des Testsystems das Programm modifizieren und Testparameter so einstellen, daß die Mittenfrequenz und die Frequenzspanne auf der Instrumententafel so angezeigt werden, daß eine Instrumententafel zur Steuerung der neuen Hardware angezeigt werden kann.

Andererseits kann der Benutzer unter Verwendung der dritten Kombination einen neuen Testparameter eines beliebigen Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung eines vom Testelement erhaltenen Ergebnisses erzeugen, einen neuen Test mit dem erzeugten Testparameter durchführen und ein vom neuen Test erhaltenes Ergebnis auf der graphischen Anzeigevorrichtung anzeigen, indem er lediglich Koordinaten auf der graphischen Darstellung des Testergebnisses mittels der Zeigevor-

richtung spezifiziert und ein Graphiksymbol des Testelementnamens auswählt. Wenn jedoch ein Testgegenstand, der als Vorgabe gestartet wird, vom Testelement nicht ermittelt werden kann, um den neuen Test zu starten, wird eine Operation zum Auswählen des Graphiksymbols des Testgegenstandes nach dem Auswählen des Graphiksymbols des Testelementnamens hinzugefügt.

Genauer, während ein Programm herkömmlicherweise modifiziert werden muß, um eine Frequenz einer unerwarteten Störung oder dergleichen zu analysieren, ermöglicht die dritte Kombination, eine solche Operation zu erhalten durch bis zu dreimaliges Betätigen der Zeigevorrichtung ohne Modifizierung des Programms insgesamt.

- (4) Kombination aus Testaufgabeneinrichtung, Testparameterspeichereinrichtung, Testdatenspeichereinrichtung, virtueller Instrumenteneinrichtung, Darstellungsblatteinrichtung und Testklasseneinrichtung, sowie Testplaneinrichtung (vierte Kombination)

Dies ist eine Kombination aus der Testplaneinrichtung und der obenbeschriebenen ersten Kombination. Wenn die Testplaneinrichtung der obenwähnten Kombination hinzugefügt wird, wird es möglich, ein Programmelement oder ein zusammengesetztes Programmelement zu kombinieren, wodurch es möglich wird, ein Programm zu schreiben und laufen zu lassen, das eine Reihenfolge für die Ausführung der Testaufgaben einstellen kann oder die Reihenfolge der Ausführung steuern kann, zusätzlich zum Erhalten der ersten, zweiten, vierten und fünften Objekte.

- (5) Kombination aus Testaufgabeneinrichtung, Testparameterspeichereinrichtung, Testdatenspeichereinrichtung, virtueller Instrumenteneinrichtung, Darstellungsblatteinrichtung und Testklasseneinrichtung, sowie Schlitzinformationseinrichtung, virtueller Instrumentendiensteneinrichtung, Wechselwirkungsrichtung und Testplaneinrichtung (fünfte Kombination)

Dies ist eine Kombination aus Schlitzinformationseinrichtung, virtueller Instrumenteneinrichtung, Wechselwirkungsrichtung und Testplaneinrichtung mit der obenbeschriebenen ersten Kombination. Die Kombination aus Testaufgabeneinrichtung, Testparameterspeichereinrichtung, Testdatenspeichereinrichtung, virtueller Instrumenteneinrichtung, Darstellungsblatteinrichtung und Testklasseneinrichtung ermöglicht, die ersten und zweiten Objekte zu erhalten, wobei das Hinzufügen der Schlitzinformationseinrichtung und der virtuellen Diensteneinrichtung ermöglicht, auch das dritte Objekt zu erhalten. Wenn außerdem die Schlitzinformationseinrichtung und die Wechselwirkungsrichtung hinzugefügt werden, können die vierten und fünften Objekte erhalten werden. Wenn außerdem die Testplaneinrichtung hinzugefügt wird, wird es möglich, ein Programmelement oder ein zusammengesetztes Programmelement zu kombinieren, wodurch eine Ausführungsreihenfolge der Testaufgaben eingestellt und die Ausführungsreihenfolge gesteuert werden kann.

Die Fig. 12 zeigt eine Anordnung eines Testsystems gemäß einer Ausführungsform, auf die die vorliegende Erfindung angewendet wird. Das Testsystem der in der Figur gezeigten Ausführungsform umfaßt einen GUI-Verarbeitungsabschnitt 100 und mehrere Testausrüstungen 110. Der GUI-Verarbeitungsabschnitt 100 dient zum Implementieren der GUI-Umgebung, die die visuelle Rückmeldung für einen Benutzer zur Verfügung stellt, und wird mittels eines Personalcomputers, eines Arbeitsplatzrechners oder einer Verarbeitungspipeline, die deren Funktionen besitzt, oder derglei-

chep implementiert. Außerdem ist der GUI-Verarbeitungsabschnitt 100 mit einem Busschnittstellenabschnitt 200 und mit mehreren Testausrüstungen 110 verbunden, die mit einem Testausrüstungsverbindungsbus 210 über den Busschnittstellenabschnitt 200 verbunden sind.

Der Testausrüstungsverbindungsbus 210 enthält z. B. ein Computernetz, wie z. B. TCP/IP, einen seriellen oder parallelen Anschluß, einen IEEE488-Anschluß oder eine Kombination hieraus.

Der GUI-Verarbeitungsabschnitt 100, wie oben beschrieben, umfaßt eine CPU 10, einen Video-RAM 12, einen Graphikprozessor 14, einen Host-Schnittstellenabschnitt 16, einen RAM-D/A-Umsetzer 18, eine Graphikanzeigevorrichtung 24, einen Speicher 26, einen Zeigevorrichtung-Schnittstellenabschnitt 28, eine Zeigevorrichtung 30, einen Zeigercursor-Anzeige/Verarbeitungsabschnitt 32 und einen GUI-Erzeugungs/Verwaltungsabschnitt 34.

Die CPU 10 ist über einen Bus mit dem Zeiger-Schnittstellenabschnitt 16, dem Speicher 26, dem Zeigervorrichtung-Schnittstellenabschnitt 28, dem Zeigercursor-Anzeige/Verarbeitungsabschnitt 32 und dem GUI-Erzeugungs/Verwaltungsabschnitt 34 verbunden und steuert den gesamten GUI-Verarbeitungsabschnitt 100. Der Video-RAM 12 speichert Bilddaten, die auf der graphischen Anzeigevorrichtung 24 angezeigt werden sollen, auf Pixelbasis. Zum Beispiel besteht der Video-RAM 12 aus einem gepackten Pixelsystem, bei welchem ein Punkt auf dem Anzeigebildschirm der Graphikanzeigevorrichtung 24 einem Pixelwert mit einer Größe von einem Byte entspricht. Das gepackte Pixelsystem verwendet üblicherweise einen Speicherbereich des Video-RAM 12, entsprechend der Anzahl der Punkte auf dem Anzeigebildschirm als ein Anzeigebereich, und einen restlichen Speicherbereich (Nichtanzeigebereich) zum Erstellen oder Speichern eines GUI-Abschnitts, wie z. B. eines Fensterbildes oder eines Graphiksymbols.

Der Graphikprozessor 14 empfängt ein von der CPU 10 gesendetes Kommando und setzt den Video-RAM 12. Außerdem erzeugt der Graphikprozessor 14 ein Synchronsignal, das für die Bildschirmanzeige erforderlich ist, liest Pixeldaten für den Anzeigebereich des Video-RAM 12 synchron mit dem erzeugten Synchronsignal und sendet diese zum RAM-D/A-Umsetzer 18. Der Speicher 26 wird für die Initialisierung des Graphikprozessors 14 oder dergleichen oder zum Speichern von Zeichensatz- oder Bilddaten verwendet.

Der RAM-D/A-Umsetzer 18 dient zum Umsetzen der vom Graphikprozessor 14 gesendeten Pixeldaten in ein Videosignal. Das umgesetzte Videosignal wird zur Graphikanzeigevorrichtung 24 gesendet. Der RAM-D/A-Umsetzer 18 enthält eine Nachschlagtabelle 20 und einen D/A-Umsetzer 22.

Außerdem wird der GUI auf dem Anzeigebildschirm der Graphikanzeigevorrichtung 24 durch die Zeigevorrichtung 30 gesteuert. Die Zeigevorrichtung 30 besitzt einen Druckknopfschalter und sendet Bewegungsinformationen sowie Ein/Aus-Informationen des Druckknopfschalters (z. B. einen Zustand, in dem der Druckknopf entsprechend einem Ein-Zustand niedergedrückt ist, während ein Zustand, in dem ein Finger oder dergleichen den Druckknopf lösläßt, einem Aus-Zustand entspricht) zur CPU 10 über den Zeigervorrichtung-Schnittstellenabschnitt 28. Der Zeigercursor-Anzeige/Verarbeitungsabschnitt 32 zeigt den Zeigercursor an einer Position auf dem Bildschirm der graphischen Anzeigevorrichtung 24 an, der durch die Zeigevorrichtung 30 angegeben wird. Genauer speichert der Zeigercursor-Anzeige/Verarbeitungsabschnitt 32 die Cursoranzeigeposition und die Position des gespeicherten Cursorbildes im Nicht-Anzeigebereich des Video-RAM 12 und überträgt das im

Nicht-Anzeigebereich des Video-RAM 12 gespeicherte Cursorbild an die Cursoranzeigeposition auf dem Anzeigebereich des Video-RAM 12, wodurch der Cursor angezeigt wird. Der Zeigercursor-Anzeige/Verarbeitungsabschnitt 32 erzeugt ein Vorgabe-Cursorbild im Nicht-Anzeigebereich des Video-RAM 12 beim Starten des Systems und speichert den Bereich des erzeugten Cursor-Bildes.

Anschließend wird die Cursor-Anzeigeposition durch Erfassen der Bewegung der Zeigervorrichtung 30 ermittelt.

Außerdem enthält der GUI-Erzeugungs/Verwaltungsabschnitt 34 einen Objekterzeugungsabschnitt 40, einen Objektverwaltungsabschnitt 42, einen Objekterfassungsabschnitt 44 und einen Ereigniserfassungs/Informierungsabschnitt 46. Der Objekterzeugungsabschnitt 40 erzeugt ein Objekt und speichert das erzeugte Objekt im Speicher 26. Der Objektverwaltungsabschnitt 42 verwaltet das Objekt im Speicher 26, das vom Objekterzeugungsabschnitt 40 erzeugt worden ist. Die Verwaltung des Objekts wird bewerkstelligt, indem im Speicher 26 eine Tabelle von Typen und Namen der Objekte und der Einsprungpunkte der Objekte gehalten wird.

Der Objekterfassungsabschnitt 44 besitzt eine Funktion zum Spezifizieren eines Typs und eines Namens eines Objekts, wodurch ein Einsprungpunkt des Objektes erfaßt wird. Diese Funktion wird implementiert durch Bezugnahme auf eine Tabelle, die der Objektverwaltungsabschnitt 42 besitzt. Außerdem besitzt der Objekterfassungsabschnitt 44 eine Funktion zum Erfassen eines Objektes, das für die Ereignisverarbeitung durch die Zeigevorrichtung 30 an beliebigen Positionskoordinaten auf dem Bildschirm der Graphikanzeigevorrichtung 24 verantwortlich ist.

Der Ereigniserfassungs/Informierungsabschnitt 46 erfaßt ein Ereignis der Zeigevorrichtung 30 und erhält Positionskoordinaten auf dem Bildschirm, an denen das Ereignis auftritt. Anschließend erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 ein Objekt zum Verwalten eines angezeigten Bildes als Antwort auf das von der Zeigevorrichtung 30 eingegebene Ereignis mit dem Objekterfassungsabschnitt 44 und führt einen Informationsprozeß über das Ereignis auf dem erfaßten Objekt durch. Das vom Ereigniserfassungs/Informierungsabschnitt 46 erfaßte Ereignis ist eines von der Zeigevorrichtung 30 und irgendeines aus der Gruppe bestehend aus Druckknopf-Niederdrücken, Druckknopf-Loslassen oder Ziehen.

Die Fig. 13 ist ein Flußdiagramm, das die Operationsprozedur eines Ereigniserfassungs/Informierungsprozesses des Ereigniserfassungs/Informierungsabschnitts 46 zeigt. Zuerst erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 einen Zustand des Druckknopfes der Zeigervorrichtung 30 (Schritt c1) und ermittelt, ob der Druckknopf niedergedrückt ist (der Zustand, in dem der Druckknopf niedergedrückt ist, ist bezeichnet mit NIEDERGEDEÜCKT) (Schritt c2). Wenn der Druckknopf nicht niedergedrückt ist, kehrt der Prozeß zum Schritt c1 zurück und wird wiederholt.

Wenn der Druckknopf niedergedrückt ist, wird ein Niederdrückungs-Abschluß-Informierungsprozeß ausgeführt (Schritt c3). Im Niederdrückungs-Abschluß-Informierungsprozeß erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 ein Objekt, für das die Ereignisverarbeitung durchgeführt wird unter Verwendung des Objekterfassungsabschnitts 44, und führt den Niederdrückungs-Abschlußprozeß für das erfaßte Objekt mit den Positionskoordinaten als Argument aus, an den das Ereignis aufgetreten ist.

Anschließend erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 die Bewegung der Zeigevorrichtung 30 (Schritt c4) und ermittelt, ob eine Bewegung ausgeführt wird (Schritt c5). Wenn die Zeigevorrichtung 30 bewegt wird, führt der Ereigniserfassungs/Informierungsabschnitt

46 einen Ziehen-Informierungsprozeß aus (Schritt c6), und erfaßt dann, wenn sie nicht bewegt wird, den Zustand des Druckknopfes der Zeigevorrichtung 30 (Schritt c7).

Im Ziehen-Informierungsprozeß in Schritt c6 erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 ein Objekt, mit dem die Ereignisverarbeitung unter Verwendung des Objekterfassungsabschnitts 44 durchgeführt wird, und veranlaßt das erfaßte Objekt, den Ziehen-Prozeß mit den Positionskordinaten als Argument auszuführen. Nachdem der Ziehen-Prozeß ausgeführt ist, rückt der Prozeß zum Schritt c7 vor, in dem der Zustand für den Druckknopf der Zeigevorrichtung 30 erfaßt wird.

Anschließend ermittelt der Ereigniserfassungs/Informierungsabschnitt 46, ob der Druckknopf sich in einem losgelassenen Zustand befindet, in dem ein Finger oder eine Hand des Benutzers den Druckknopf losläßt (dieser Zustand ist bezeichnet mit LOSGELASSEN) auf der Grundlage des Ergebnisses der Erfassung des Zustands des Druckknopfes der Zeigevorrichtung 30 in Schritt c7 (Schritt c8). Wenn er sich nicht im losgelassenen Zustand befindet, kehrt der Prozeß zum obenbeschriebenen Schritt c4 zurück und wird wiederholt. Wenn er sich außerdem im losgelassenen Zustand befindet, ermittelt der Ereigniserfassungs/Informierungsabschnitt 46, ob der Ziehen-Prozeß durchgeführt wird (Schritt c9). Ob der Ziehen-Prozeß durchgeführt wird, kann daraus ermittelt werden, ob die Zeigevorrichtung 30 in einem Zustand bewegt wird, in dem der Druckknopf niedergedrückt ist. Wenn die Zeigevorrichtung in einem Zustand, in dem der Druckknopf niedergedrückt ist, bewegt wird, wird festgestellt, daß der Ziehen-Prozeß durchgeführt wird, wobei der Ziehen-Abschluß-Informierungsprozeß durchgeführt wird (Schritt c10). Wenn andererseits kein Ziehen-Prozeß durchgeführt wird, wird ein Loslassen-Abschluß-Informierungsprozeß durchgeführt (Schritt c11).

Im Ziehen-Abschluß-Informierungsprozeß im Schritt c10 erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 ein Objekt, mit dem die Ereigniserfassung durchgeführt wird, unter Verwendung des Objekterfassungsabschnitts 44 und veranlaßt das erfaßte Objekt, den Ziehen-Abschluß-Informierungsprozeß mit den Positionskordinaten des Zeigerursors als Argument auszuführen. Außerdem erfaßt der Ereigniserfassungs/Informierungsabschnitt 46 im Losgelassen-Abschluß-Informierungsprozeß im Schritt c11 ein Objekt, mit dem die Ereignisverarbeitung durchgeführt wird, unter Verwendung des Objekterfassungsabschnitts 44, und veranlaßt das erfaßte Objekt, den Losgelassen-Abschluß-Informierungsprozeß mit den Positionskordinaten des Zeigerursors, an denen das Loslassen aufgetreten ist, als Argument auszuführen. Anschließend kehrt der Prozeß für eine Wiederholung zum obenbeschriebenen Schritt c1 zurück.

Anschließend besitzt jedes Objekt eine "Tabelle der dem Ereignis zugeordneten Prozesse", in denen Prozesse definiert sind, die gestartet werden durch den Niederdrückungs-Prozeß, Ziehen-Prozeß, Ziehen-Abschluß-Prozeß und Loslassen-Abschluß-Prozeß. Wenn jeder Prozeß ausgeführt worden ist, wird anschließend auf den vorherigen Satz "Tabelle der dem Ereignis zugeordneten Prozesse" zugegriffen, um einen einem aufgetretenen Ereignis zugeordneten Prozeß auszuführen.

Im folgenden wird auf der Grundlage der jeweiligen Ausführungsform eine Übersicht des Objekterzeugungsabschnitts 40 beschrieben.

#### [Ausführungsform 1]

Die Ausführungsform 1 ist ein Fall, der einen Objekterzeugungsabschnitt 40 verwendet, der Funktionen der Testparameterspeichereinrichtung, der Testdatenspeicherein-

richtung, der virtuellen Instrumenteneinrichtung, der virtuellen Instrumentendiensteinrichtung, der Darstellungsblatteinrichtung, der Testklasseneinrichtung, der Schlitzinformationseinrichtung, der Wechselwirkungseinrichtung und der Testplaneinrichtung besitzt. Im folgenden wird ein Stamm-Objekt, eine Zuordnungsbeziehung, ein Attribut und ein Prozeß eines Objektes genauer beschrieben, welches von jeweils der Testparameterspeichereinrichtung, der Testdatenspeichereinrichtung, der virtuellen Instrumenteneinrichtung, der virtuellen Instrumentendiensteinrichtung, der Darstellungsblatteinrichtung, der Testklasseneinrichtung, der Schlitzinformationseinrichtung, der Wechselwirkungseinrichtung und der Testplaneinrichtung verwendet wird.

#### Testgegenstand

[Stamm-Objekt]: Keines

[Zuordnungsbeziehung]

#### Testdateiname

Er enthält mehrere Testbedingungsdateinamen, die der Testgegenstand nutzen kann.

[Attribut]

#### Name

Er ist ein Bereich zum Speichern einer Zeichenkette, die einen Namen des Testgegenstandes im Speicher 26 darstellt.

#### Anzeigebild

Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols des Testgegenstandes im Speicher 26.

[Prozeß]

Erzeuge Graphiksymbol- (Positionskordinaten, Anzeigebereich)

Er greift auf das Anzeigebild-Attribut zu, um das Graphiksymbol des Testgegenstandes an den Positionskordinaten, die von einem Argument geliefert werden, auf dem Anzeigebereich anzuzeigen, der von einem Argument geliefert wird.

Zeige Graphiksymbol des verfügbaren Testbedingungsdateinamens an (Anzeigebereich)

Er zeigt ein Graphiksymbol des Testbedingungsdateinamens auf einem Anzeigebereich an, der von einem Argument geliefert wird, unter Verwendung des Erzeuge-Graphiksymbol-Prozesses des Testbedingungsdateinamens, der diesen zuordnet.

Testbedingungsdateiname

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]

Name

Dies ist ein Bereich zum Speichern des Namens des Testbedingungsdateinamens, der im Speicher 26 gespeichert ist, im Speicher 26.

Anzeigebild

Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols des Testbedingungsdateinamens im Speicher 26.

[Prozeß]

Erzeuge Graphiksymbol (Positionskordinaten, Anzeigebereich)

Er greift auf das Anzeigebild-Attribut zu, um das Graphiksymbol des Testbedingungsdateinamens an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

Testkompromiß

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]

Kompromißinformation

Dies ist ein Bereich zum Speichern der Testbedingungsinformation, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt werden soll, im Speicher als Zeichenkette.

Positionskordinaten des Graphiksymbols des Testelementnamens, der dem Hochgeschwindigkeitstest zugeordnet ist

Dies ist ein Bereich zum Speichern der Positionskordinaten eines Graphiksymbols des Testelementnamens, der einem Hochgeschwindigkeitstest zugeordnet ist, im Speicher 26.

Positionskordinaten des Graphiksymbols des Testelementnamens, der dem hochgenauen Test zugeordnet ist

Dies ist ein Bereich zum Speichern der Positionskordinaten eines Graphiksymbols des Testelementnamens, der einem hochgenauen Test zugeordnet ist, im Speicher 26.

[Prozeß]

Zeige Graphiksymbol an (Anzeigebereich auf Graphikanzeigevorrichtung, Einsprungpunkt des Testelementnamens)

Dies erzeugt Graphiksymbole des Testelementnamens,

die einem Hochgeschwindigkeitstest und einem hochgenauen Test zugeordnet sind, auf dem von einem Argument gelieferten Anzeigebereich auf der Graphikanzeigevorrichtung. Das Bild des zu erzeugenden Graphiksymbols bezieht sich auf das Anzeigebild-Attribut des von einem Argument gelieferten Testelementnamens. Die Positionskordinaten des erzeugten Graphiksymbols des Testelementnamens, der dem Hochgeschwindigkeitstest und dem hochgenauen Test zugeordnet ist, auf dem vom Argument gelieferten Anzeigebereich auf der Graphikanzeigevorrichtung werden jeweils gespeichert in den Attributen "Positionskordinaten des Graphiksymbols des Testelementnamens, der dem Hochgeschwindigkeitstest zugeordnet ist" und "Positionskordinaten des Graphiksymbols des Testelementnamens, der dem hochgenauen Test zugeordnet ist".

Setze Kompromiß (Positionskordinaten)

Dies greift auf die Attribute "Positionskordinaten des Graphiksymbols des Testelementnamens, der dem Hochgeschwindigkeitstest zugeordnet ist" und "Positionskordinaten des Graphiksymbols des Testelementnamens, der dem hochgenauen Test zugeordnet ist" zu. Wenn anschließend das Graphiksymbol an den Positionskordinaten, die vom Argument geliefert werden, das Graphiksymbol des Testelementnamens ist, der dem Hochgeschwindigkeitstest zugeordnet ist, speichert es die Zeichenkette der Hochgeschwindigkeitsmessung im Kompromißinformations-Attribut. Wenn das Graphiksymbol an den Positionskordinaten, die vom Argument geliefert werden, das Graphiksymbol des Testelementnamens ist, der dem hochgenauen Test zugeordnet ist, speichert es die Zeichenkette des hochgenauen Tests im Kompromißinformations-Attribut.

Testziel

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]

Testgegenstand

Dies ordnet einen Testgegenstand zu.

Testbedingungsdateiname

Dies ordnet einen Testbedingungsdateinamen und einen Testkompromiß zu.

[Attribut]: keines

[Prozeß]

Setze Testgegenstand (Einsprungpunkt von Testgegenstand)

Er setzt einen Einsprungpunkt des Testgegenstands, der von einem Argument geliefert wird, in den Zuordnungszustand.

Setze Testbedingungsdateiname (Einsprungpunkt von Testbedingungsdateiname)

Er setzt einen Einsprungpunkt des Testbedingungsdateinamens, der von einem Argument geliefert wird, in Zuordnungsbeziehung.

Setze Testkompromiß (Einsprungpunkt von Testkompromiß)

Er setzt einen Einsprungpunkt des Testkompromisses, der von einem Argument geliefert wird, in Zuordnungsbeziehung.

Testelementname

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]

Testgegenstand

Dies ordnet mehrere Testgegenstände zu, die in einem Testelement verwendet werden können.

[Attribut]

Name

Dies ist ein Bereich zum Speichern einer Zeichenkette, die einen Testelementnamen darstellt, im Speicher 26.

Anzeigebild

Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols des Testelementnamens im Speicher 26.

Testgegenstand durch Vorgabe gestartet

Dies ist ein Bereich zum Speichern eines Einsprungpunktes eines Testgegenstandes, der durch Vorgabe gestartet wird.

[Prozeß]

Erzeuge Graphiksymbol (Positionskoordinaten, Anzeigebereich)

Dies greift auf das Anzeigebild-Attribut zu, um das Graphiksymbol des Testelementnamens an den von einem Argument gelieferten Positionskoordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

Zeige Graphiksymbol des verfügbaren Testgegenstandes an (Anzeigebereich)

Dies erzeugt ein Graphiksymbol des Testgegenstandes auf dem von einem Argument gelieferten Anzeigebereich unter Verwendung des Erzeuge-Graphiksymbol-Prozesses des Testgegenstandes, den es zuordnet.

Testaufgabe

[Stamm-Objekt]: Aufgabenkomponente

[Zuordnungsbeziehung]

Testziel

Dies ordnet mehrere Testziele zu.

Testelementname

Dies ordnet einen Testelementnamen zu.

[Attribut]

Name

5 Dies ist ein Bereich zum Speichern einer Zeichenkette, die eine Testaufgabe darstellt, im Speicher 26.

Anzeigebild der Testaufgabe

10 Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols einer Testaufgabe im Speicher 26.

DUT-Anzeigebild

15 Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols einer DUT im Speicher 26.

Basisentwurf einer Testaufgabe

20 Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf der graphischen Anzeigevorrichtung, den die Testaufgabe verwendet, im Speicher 26.

Entwurf der Testaufgabe

25 Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Basisentwurf der Testaufgabe, den die Testaufgabe benutzt, im Speicher 26.

30 Testgegenstandentwurf

Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Testaufgabenentwurf zum Anzeigen eines verfügbaren Testgegenstandes im Speicher 26.

Testbedingungsentwurf

Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Testaufgabenentwurf zum Anzeigen eines verfügbaren Testbedingungsdateinamens im Speicher 26.

Arbeitsraum

45 Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Testaufgabenentwurf für ein DUT-Graphiksymbol, von Graphiksymbolen der Testelementnamen, die dem Hochgeschwindigkeitstest und dem hochgenauen Test zugeordnet sind, eines Graphiksymbols des ausgewählten Testgegenstandes und eines Graphiksymbols des ausgewählten Testbedingungsdateinamens im Speicher 26.

Testelementanzeige-Entwurf

55 Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Basisentwurf der Testaufgabe zum Anzeigen eines verfügbaren Testgegenstandes im Speicher 26.

[Prozeß]

60 Initialisiere Testaufgabe ()

Der initialisiere-Testaufgabe-Prozeß führt die folgenden Prozesse aus:

65 1) Ausführen des Prozesses "erzeuge Anzeigebereich";  
2) Erfassen einer Wechselwirkungsvorrichtung mit der Objekterfassungseinrichtung und Anzeigen eines



Graphiksymbol des Testelementnamens, der einem virtuellen Instrument zugeordnet ist, das die Wechselwirkungsverrichtung zuordnet, auf einem von einem Argument gelieferten Anzeigebereich;

3) Ausführen des Erzeuge-Graphiksymbol-Prozesses durch Spezifizieren eines Anzeigebereichs, der im Attribut Testelementanzeige-Entwurf gespeichert ist; und

4) Ausführen des Erzeuge-DUT-Graphiksymbol-Prozesses durch Spezifizieren eines Anzeigebereichs, der im Attribut Arbeitsraum gespeichert ist, und durch Spezifizieren der Anzeigekoordinaten.

#### Erzeuge Anzeigebereich ()

Dies erzeugt Anzeigebereiche entsprechend den Attributen Testaufgaben-Basistentwurf, Testaufgaben-Entwurf, Testelement-Entwurf, Arbeitsraum und Testelement-Entwurf auf der graphischen Anzeige und speichert diese im jeweiligen Attribut.

#### Erzeuge Graphiksymbol (Positionskordinaten, Anzeigebereich)

Dies erzeugt das Attribut Testaufgaben-Anzeigebild, um das Graphiksymbol der Testaufgabe an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

#### Erzeuge DUT-Graphiksymbol (Positionskordinaten, Anzeigebereich)

Dies greift auf das Attribut DUT-Anzeigebild zu, um das DUT-Graphiksymbol an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

#### Wähle Testelementname (Einsprungpunkt von Testelementname)

Der Prozeß Wähle-Testelementname ist ein Prozeß, der ausgeführt wird, wenn ein Graphiksymbol des Testelementnamens auf einen Bereich fallengelassen wird, der durch das Attribut Arbeitsraum angegeben ist. Der Prozeß Wähle-Testelementname setzt zuerst einen Einsprungpunkt des von einem Argument gelieferten Testelementnamens in Zuordnungsbeziehung. Anschließend erzeugt er den Testkompromiß und führt den setze-Testkompromiß-Prozeß des Testziels aus durch Spezifizieren eines Einsprungpunktes des erzeugten Testkompromisses. Anschließend führt er den Graphiksymbolanzeige-Prozeß des erzeugten Testkompromisses mit dem im Attribut Arbeitsraum als Argument gespeicherten Anzeigebereich aus. Schließlich führt er den Prozeß "zeige Graphiksymbol des verfügbaren Testgegenstandes" des von einem Argument gelieferten Testelementnamens mit dem im Attribut Testbedingungen-Entwurf als Argument gespeicherten Anzeigebereich aus.

#### Wähle-Testgegenstand (Einsprungpunkt des Testgegenstands)

Der Prozeß wähle-Testgegenstand ist ein Prozeß, der ausgeführt wird, wenn ein Graphiksymbol des Testgegenstandes auf einen Bereich fallengelassen wird, der durch das Attribut Arbeitsraum angegeben ist. Der Prozeß Wähle-Testgegenstand führt zuerst den Prozeß Setze-Testgegenstand des Testziels aus, das die Testaufgabe zuordnet, durch Spezifizieren eines Einsprungpunktes des von einem Argument

gelieferten Testgegenstandes. Anschließend wählt er ein Graphiksymbol des gewählten Messungsgegenstandes auf dem Testgegenstand-Entwurf. Anschließend führt er den Prozeß Erzeuge-Graphiksymbol des von einem Argument gelieferten Testgegenstandes aus durch Spezifizieren eines Anzeigebereichs, der das Attribut Arbeitsraum speichert, und der Positionskordinaten. Schließlich zeigt er das Graphiksymbol des verfügbaren Testbedingungen-Testelementnamens des von einem Argument gelieferten Testgegenstandes an durch Spezifizieren eines Anzeigebereichs, der im Attribut Testbedingungen-Entwurf gespeichert ist.

#### Wähle-Testbedingungendateiname (Einsprungpunkt von Testbedingungendateiname)

Der Prozeß Wähle-Testbedingungendateiname ist ein Prozeß, der ausgeführt wird, wenn ein Graphiksymbol des Testbedingungendateinamens auf einen Bereich fallengelassen wird, der vom Attribut Arbeitsraum angegeben wird. Der Prozeß Wähle-Testbedingungendateiname führt zuerst den Prozeß Setze-Testbedingungendateiname des Testziels aus, das die Testaufgabe zuordnet, durch Spezifizieren eines Einsprungpunktes des von einem Argument gelieferten Testbedingungendateinamens. Anschließend löscht er ein Graphiksymbol des ausgewählten Testbedingungendateinamens auf dem Testbedingungen-Entwurf. Schließlich führt er den Erzeuge-Graphiksymbol-Prozeß des von einem Argument gelieferten Testbedingungendateinamens aus durch Spezifizieren eines Anzeigebereichs, der im Attribut Arbeitsraum gespeichert ist, und der Positionskordinaten.

#### Fallenlassen-Prozeß des DUT-Graphiksymbols (Positionskordinaten)

Der Prozeß des Fallenlassens des DUT-Graphiksymbols ist ein Prozeß, der ausgeführt wird, wenn ein Graphiksymbol der DUT auf einem Bereich fallengelassen wird, der vom Attribut Arbeitsraum angegeben wird. Der Prozeß des Fallenlassens des DUT-Graphiksymbols führt die folgenden Prozesse nur dann aus, wenn Graphiksymbole des Testelementnamens, die dem Hochgeschwindigkeitstest und dem hochgenauen Test zugeordnet sind, an den von einem Argument gelieferten Positionskordinaten vorhanden sind. Zuerst erfaßt er den Testkompromiß, den das Testziel zuordnet, vom Testziel, das die Testaufgabe zuordnet, und führt den Setze-Testkompromiß-Prozeß des Testkompromisses aus durch Spezifizieren der von einem Argument gelieferten Positionskordinaten. Anschließend führt er den Setze-Kompromiß-Prozeß des Prozesses "Verbinde DUT-Graphiksymbol und Testelementname-Graphiksymbol" aus durch Spezifizieren der von einem Argument gelieferten Positionskordinaten. Schließlich startet er den Ausführungsprozeß.

#### Verbinde DUT-Graphiksymbol und Testelementname-Graphiksymbol (Positionskordinaten)

Dies verbindet entweder eines der Graphiksymbole des Testelementnamens, das dem Hochgeschwindigkeitstest oder dem hochgenauen Test zugeordnet ist, an den durch ein Argument spezifizierten Positionskordinaten mit dem DUT-Graphiksymbol.

#### Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Ausführungsprozeß erfaßt zuerst ein virtuelles Instrument, das einem Testelementnamen zugeordnet ist. Anschließend führt er den Teststartprozeß des erfaßten virtuel-

len Instruments aus durch Spezifizieren des Testziels, das dieses zuordnet, und eines Einsprungpunkts der Zeichenkette des Testergebnisses, das dem von einem Argument gelieferten Testgegenstand zugeordnet ist.

Testparametergegenstand

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]

Testparametergegenstandsname

Dies ist ein Bereich zum Speichern einer Zeichenkette, die einen Namen des Testparametergegenstandes darstellt, im Speicher 26.

Parameter

Dies ist ein Bereich zum Speichern eines Parameters des Testparametergegenstands im Speicher 26.

[Prozeß]: keiner

Testparameter

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]

Testparametergegenstand

Dies ordnet mehrere Testparametergegenstände zu, die für das virtuelle Instrument erforderlich sind, um den Test einmal durchzuführen.

Testdaten

Dies ordnet einen Satz Testdaten zu zum Speichern der unter Verwendung des Testparameters erhaltenen Daten.

[Attribut]: keines

[Prozeß]

Anbringung (Einsprungpunkt des Testparametergegenstandes)

Dies ist ein Prozeß zum Setzen des Testparametergegenstands, der von einem Argument geliefert wird, in Zuordnungsbeziehung.

Testdaten

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]

Testparameter

Dies ordnet einen Testparameter zu, der beim Aufnehmen der Testdaten verwendet wird.

[Attribut]

X-Achsen-Daten

5 Dies ist ein Bereich zum Speichern der Matrixdaten z. B. des Gleitkomma-Typs, die vom Testelement erhalten werden, im Speicher 26.

Y-Achsen-Daten

10 Dies ist ein Bereich zum Speichern der Matrixdaten z. B. des Gleitkomma-Typs, die vom Testelement erhalten werden, im Speicher 26.

[Prozeß]

Nehme X-Achsen-Daten auf (Matrixposition)

20 Dies ist ein Prozeß zum Lesen der Daten des Attributs X-Achsen-Daten an einer von einem Argument gelieferten Matrixposition.

Nehme Y-Achsen-Daten auf (Matrixposition)

25 Dies ist ein Prozeß zum Lesen der Daten des Attributs Y-Achsen-Daten an einer von einem Argument gelieferten Matrixposition.

Virtuelles Instrument

30 [Stamm-Objekt]: virtuelles Instrument

[Zuordnungsbeziehung]

Testdaten

Dies ordnet mehrere Testdaten zu, die als Ergebnis des Tests erhalten werden.

40 Testparameter

Dies ordnet mehrere für den Test verwendete Testparameter zu.

45 Darstellungsblatt

Dies ordnet mehrere Darstellungsblätter zu zum Anzeigen der Testdaten auf der Graphikanzeigevorrichtung.

50 [Attribut]: keines

[Prozeß]

55 Starte Messung (Einsprungpunkt des Testziels, Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Prozeß starte-Messung führt die folgenden Prozesse aus:

- 60 1) Erzeuge einen Testparameter und spezifiziere den erzeugten Testparametergegenstand im Einsprungpunkt des Testparametergegenstands, um den Prozeß Anbringung des erzeugten Testparameters auszuführen.
- 65 2) Erzeuge alle Testparametergegenstände, die das virtuelle Instrument für einen Test benötigt, und spezifiziere den erzeugten Testparametergegenstand im Ein-

sprungpunkt des Testparametergegenstands, um den Prozeß Anbringung des erzeugten Testparameters auszuführen.

3) Erzeuge eine Testklasse, die dem Testgegenstand entspricht, den das von einem Argument gelieferte Testziel zuordnet.

4) Spezifiziere einen Einsprungpunkt des virtuellen Instruments, um den Prozeß Initialisierung der erzeugten Testklasse auszuführen.

5) Wenn das von einem Argument gelieferte Testziel einen Testbedingungsdateinamen zuordnet, spezifiziere einen Einsprungpunkt des Testbedingungsdateinamens, den das Testziel zuordnet, und führe den Prozeß Setze-Testbedingung der erzeugten Testklasse aus.

6) Spezifiziere die Einsprungpunkte der erzeugten Testparameter und des virtuellen Instruments und führe den Prozeß Erzeuge-Testparameter der erzeugten Testklasse aus.

7) Spezifiziere einen Einsprungpunkt des erzeugten Testbedingungsparameters und führe den Prozeß Erzeuge-Testdaten aus, um Testdaten zu erzeugen.

8) Spezifiziere einen Einsprungpunkt der erzeugten Testdaten und führe den Prozeß Erzeuge-Darstellungsblatt aus, um ein Darstellungsblatt zu erzeugen.

9) Spezifiziere einen Einsprungpunkt des erzeugten Testparameters und führe den Prozeß Testausführung aus.

10) Spezifiziere einen Satz von Einsprungpunkten der erzeugten Testdaten und einen Einsprungpunkt einer Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist, und führe den Prozeß "erhalte Testergebnis, das dem Testgegenstand zugeordnet ist" der erzeugten Testklasse aus.

Erzeuge Testparameter mit Testgegenstand und Schlitzinformation (Einsprungpunkt des Testgegenstands, Einsprungpunkt der Einsprunginformation der Schlitzinformation)

Der Prozeß Erzeuge-Testparameter führt die folgenden Prozesse aus:

- 1) Erzeuge Testparameter und setze diesen in Zuordnungsbeziehung des virtuellen Instruments.
- 2) Erzeuge alle Testparametergegenstände, die das virtuelle Instrument für einen Test benötigt, und spezifiziere den erzeugten Testparametergegenstand im Einsprungpunkt des Testparametergegenstands, um den Prozeß Anbringung des erzeugten Testparameters auszuführen.
- 3) Erzeuge eine Testklasse, die dem von einem Argument gelieferten Testgegenstand zugeordnet ist.
- 4) Spezifiziere einen Einsprungpunkt des virtuellen Instruments, um den Prozeß Initialisierung der erzeugten Testklasse auszuführen.
- 5) Spezifiziere einen Einsprungpunkt der von einem Argument gelieferten Schlitzinformation und führe den Prozeß Setze-Schlitzinformation der erzeugten Testklasse aus.
- 6) Spezifiziere die Einsprungpunkte des erzeugten Testparameters und des virtuellen Instruments und führe den Prozeß Erzeuge-Testparameter der erzeugten Testklasse aus.
- 7) Spezifiziere einen Einsprungpunkt des erzeugten Testparameters und führe den Prozeß Erzeuge-Testdaten aus, um Testdaten zu erzeugen.
- 8) Spezifiziere einen Einsprungpunkt der erzeugten

Testdaten und führe den Prozeß Erzeuge-Darstellungsblatt aus, um ein Darstellungsblatt zu erzeugen.

#### 5 Erzeuge Testdaten (Einsprungpunkt von Testparameter)

Der Prozeß Erzeuge-Testdaten erzeugt zuerst Testdaten. Anschließend setzt er die erzeugten Testdaten in Zuordnungsbeziehung des virtuellen Instruments. Anschließend setzt er den von einem Argument gelieferten Testparameter in Zuordnungsbeziehung der erzeugten Testdaten. Schließlich setzt er die erzeugten Testdaten in Zuordnungsbeziehung des von einem Argument gelieferten Testparameters.

#### 15 Erzeuge Darstellungsblatt (Einsprungpunkt von Testdaten)

Der Prozeß Erzeuge-Darstellungsblatt erzeugt ein Darstellungsblatt und setzt die von einem Argument gelieferten Testdaten in Zuordnungsbeziehung des erzeugten Darstellungsblattes.

#### Testausführung (Einsprungpunkt von Testparametern)

Der Prozeß Testausführung setzt den Testparametergegenstand, der den von einem Argument gelieferten Testparameter im Testelement zuordnet, führt die Messung durch durch Steuern des Testelements und erhält die Testdaten vom Testelement. Die erhaltenen Testdaten werden in Bereichen im Speicher der Attribute X-Achsen-Daten und Y-Achsen-Daten der Testdaten gespeichert, die der von einem Argument gelieferte Testparameter zuordnet. Anschließend erhält er ein Darstellungsblatt, das die Testdaten zuordnet, und implementiert den Aktualisierungsprozeß des erfaßten Darstellungsblattes.

#### Setze Testparametergegenstand (Daten, Testparametergegenstandsname, Einsprungpunkt des Testparameters)

Der Prozeß Setze-Testparametergegenstand erfaßt zuerst einen Testparametergegenstand, dessen von einem Argument gelieferter Testparametergegenstandsname das Attribut Name des Testparametergegenstands des Testparametergegenstands ist, der vom von einem Argument gelieferten Testparameter gehalten wird. Anschließend setzt er die von einem Argument gelieferten Daten im Attribut Parameter des erfaßten Testparametergegenstands.

#### Virtueller Instrumentendienst

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]

Anzeigebild

Dies ist ein Bereich zum Speichern eines Anzeigebildes des Graphiksymbols des virtuellen Instrumentendienstes im Speicher.

[Prozeß]

Dienstaussführung (Einsprungpunkt von Schlitzinformation) Dies ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Dienstaussführung für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt, wie z. B. die Spitzenwertsuche, als virtueller Instru-

mentendienst behandelt wird.

Erzeuge Graphiksymbol (Positionskoordinaten, Anzeigebereich)

Dies erzeugt das Attribut Anzeigebild, um das Graphiksymbol des virtuellen Instrumentendienstes an den von einem Argument gelieferten Positionskoordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

Spitzenwertsuche

[Stamm-Objekt]

Virtueller Instrumentendienst

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Dienstaussführung (Einsprungpunkt von Schlitzinformation)

Der Prozeß Dienstaussführung nimmt zuerst die Matrixdaten der X- und Y-Achsen in einem spezifizierten Bereich, der von Schlitzinformation geliefert wird, aus den Testdaten auf, die von der von einem Argument gelieferten Schlitzinformation gehalten werden. Anschließend erfaßt er eine Matrixnummer mit einem Spitzenwert von den aufgenommenen X-Achsen-Matrixdaten. Anschließend stellt er die Daten der X- und Y-Achsen der Matrixnummer mit dem Spitzenwert auf der graphischen Darstellung dar. Schließlich zeigt er ein Graphiksymbol, das den Spitzenwert darstellt, an einer Position auf der graphischen Darstellung an, die der Matrixnummer mit dem Spitzenwert entspricht.

Darstellungsblatt

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]

Virtuelles Instrument

Dies ordnet ein virtuelles Instrument zu, das das Darstellungsblatt erzeugt.

Testdaten

Dies ordnet Testdaten zu, die die Testdaten enthalten, die graphisch dargestellt werden sollen.

Schlitzinformation

Dies ordnet mehrere Schlitzinformationen zu, aus denen das Darstellungsblatt erzeugt wird.

[Attribut]: keines

[Prozeß]

Testergebnisanzeige ()

Der Prozeß Testergebnisanzeige führt die folgenden Prozesse aus:

- 1) Erzeuge einen Bereich auf der graphischen Anzeigevorrichtung zum Anzeigen des Testergebnisses.
- 2) Zeige einen Graph an durch Zugreifen auf die Testdaten, die dieser zuordnet.

Aktualisierung ()

Der Prozeß Aktualisierung löscht einen angezeigten Graphen und zeigt einen Graphen erneut an durch Zugreifen auf die Testdaten, die dieser zuordnet.

Hervorhebung (Minimalwert auf X-Achse, Maximalwert auf X-Achse, Minimalwert auf Y-Achse, Maximalwert auf Y-Achse)

Der Prozeß Hervorhebung ist ein Prozeß, der von dem Ereigniserfassung/Informierungsabschnitt 46 ausgeführt wird, wenn ein Bereich durch eine Operation der Zeigevorrichtung 30 auf der graphischen Darstellung, die vom Darstellungsblatt erzeugt wird, spezifiziert wird. Der Prozeß Hervorhebung führt die folgenden Prozesse aus:

- 1) Spezifiziere den von einem Argument gelieferten Minimalwert auf der X-Achse, den Maximalwert auf der X-Achse, den Minimalwert auf der Y-Achse und den von einem Argument gelieferten Maximalwert auf der Y-Achse und führe den Prozeß Erzeuge-Schlitzinformation aus, um die Schlitzinformation zu erzeugen.
- 2) Führe den Prozeß Aktualisierung der erzeugten Schlitzinformation aus.
- 3) Führe den Prozeß "zeige verfügbaren virtuellen Instrumentendienst an" aus.
- 4) Führe den Prozeß "zeige verfügbaren Testelementnamen an" aus.

Erzeuge Schlitzinformation (Minimalwert auf X-Achse, Maximalwert auf X-Achse, Minimalwert auf Y-Achse, Maximalwert auf Y-Achse)

Einzelheiten des Prozesses Erzeuge-Schlitzinformation sind im folgenden gezeigt:

- 1) Erzeuge die Schlitzinformation und setze einen Einsprungpunkt des Darstellungsblattes selbst in das Attribut Darstellungsblatt der Schlitzinformation ein.
- 2) Führe den Prozeß "nehme X-Achsen-Daten auf, die den Positionskoordinaten auf der X-Achse zugeordnet sind" aus, durch Spezifizieren des von einem Argument gelieferten Minimalwertes auf der X-Achse, und nehme die X-Achsen-Daten auf, die dem Minimalwert auf der X-Achse zugeordnet sind.
- 3) Führe den Prozeß "nehme X-Achsen-Daten auf, die den Positionskoordinaten auf der X-Achse zugeordnet sind" aus durch Spezifizieren des von einem Argument gelieferten Maximalwertes auf der X-Achse und nehme die X-Achsen-Daten auf, die dem Maximalwert auf der X-Achse zugeordnet sind.
- 4) Führe den Prozeß "nehme Y-Achsen-Daten auf, die den Positionskoordinaten auf der Y-Achse zugeordnet sind" aus, durch Spezifizieren des von einem Argument gelieferten Minimalwertes auf der Y-Achse, und nehme die Y-Achsen-Daten auf, die dem Minimalwert auf der Y-Achse zugeordnet sind.
- 5) Führe den Prozeß "nehme Y-Achsen-Daten auf, die den Positionskoordinaten auf der Y-Achse zugeordnet sind" aus durch Spezifizieren des von einem Argument

gelieferten Maximalwerts auf der Y-Achse und nehme die Y-Achsen-Daten auf, die dem Maximalwert auf der Y-Achse zugeordnet sind.

6) Spezifiziere die dem Minimalwert auf der X-Achse zugeordneten X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordneten X-Achsen-Daten, die dem Minimalwert auf der Y-Achse zugeordneten Y-Achsen-Daten und die dem Maximalwert auf der Y-Achse zugeordneten Y-Achsen-Daten, um den Prozeß Initialisierung der erzeugten Schlitzinformation auszuführen.

Zeige verfügbaren virtuellen Instrumentendienst an (Anzeigebereich)

Der Prozeß "zeige verfügbaren virtuellen Instrumentendienst an" greift auf virtuelle Instrument zu, den er zuordnet, um ein Graphiksymbol des virtuellen Instrumentendienstes, das die virtuelle Instrument zuordnet, auf einem Argument gelieferten Anzeigebereich anzuzeigen.

Zeige verfügbaren Testelementnamen an (Anzeigebereich)

Der Prozeß "zeige verfügbaren Testelementnamen an" erfaßt eine Wechselwirkungsvorrichtung mit der Objekterfassungseinrichtung und zeigt ein Graphiksymbol des Testelementnamens, der dem virtuellen Instrument zugeordnet ist, das die Wechselwirkungsvorrichtung zuordnet, auf einem von einem Argument gelieferten Anzeigebereich an.

Nehme X-Achsen-Daten auf, die die Positionskoordinaten auf der X-Achse zugeordnet sind (Positionskoordinaten)

Der Prozeß "Nehme X-Achsen-Daten auf, die den Positionskoordinaten auf der X-Achse zugeordnet sind" ist ein Prozeß, der auf die Testdaten zugreift, die er zuordnet, und der die X-Achsen-Daten aufnimmt, die den von einem Argument gelieferten Positionskoordinaten zugeordnet sind.

Nehme Y-Achsen-Daten auf, die die Positionskoordinaten auf der Y-Achse zugeordnet sind (Positionskoordinaten)

Der Prozeß "Nehme Y-Achsen-Daten auf, die den Positionskoordinaten auf der Y-Achse zugeordnet sind" ist ein Prozeß, der auf die Testdaten zugreift, die er zuordnet, und der die Y-Achsen-Daten aufnimmt, die den von einem Argument gelieferten Positionskoordinaten zugeordnet sind.

Testklasse

[Stamm-Objekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Initialisierung (Einsprungpunkt des virtuellen Instruments)

Der Prozeß Initialisierung der Testklasse bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Initialisierung für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Testklasse behandelt wird.

Setze Testbedingung (Einsprungpunkt des Testbedingungsdatensamens)

Der Prozeß Setze-Testbedingung der Testklasse bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Setze-Testbedingung für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Testklasse behandelt wird.

Setze Schlitzinformation (Einsprungpunkt der Schlitzinformation)

Der Prozeß Setze-Schlitzinformation der Testklasse bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Setze-Schlitzinformation für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Testklasse behandelt wird.

Erzeuge-Testparameter (Einsprungpunkt des Testparameters)

Der Prozeß Erzeuge-Testparameter bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Erzeuge-Testparameter für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Testklasse behandelt wird.

Nehme Testergebnis auf, das dem Testgegenstand zugeordnet ist (Einsprungpunktsatz der Testdaten, Einsprungpunkt der Kette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Prozeß "Nehme Testergebnis auf, das dem Testgegenstand zugeordnet ist" bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Erzeuge-Testparameter für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Testklasse behandelt wird.

Blatttestklasse

[Stamm-Objekt]: Testklasse

[Zuordnungsbeziehung]

Implementierungsvorrichtung der Testklasse

Dies ordnet eine Implementierungsvorrichtung der Testklasse zu, die einen abstrakten Testparameter, der von der Blatttestklasse erzeugt worden ist, in einen spezifischen Testparameter für jedes virtuelle Instrument umsetzt.

[Attribut]: keines

[Prozeß]

Erzeuge Testparameter (Einsprungpunkt des Testparameters)

Dies spezifiziert einen Einsprungpunkt des von einem Argument gelieferten Testparameters und führt den Prozeß Erzeuge-Testparameter der Implementierungsvorrichtung der Testklasse aus, die diesen zuordnet.

## Leistungsspektrummessung

[Stamm-Objekt]: Blatttestklasse

[Zuordnungsbeziehung]: keine

[Attribut]

Mittenfrequenz

Dies ist ein Bereich zum Speichern der Mittenfrequenz für die Messung des Leistungsspektrums im Speicher 26.

Frequenzspanne

Dies ist ein Bereich zum Speichern einer Frequenzspanne zur Messung eines Leistungsspektrums im Speicher 26.

[Prozeß]

Initialisierung (Einsprungpunkt des virtuellen Instruments)

Der Prozeß Initialisierung erzeugt eine Implementierungsvorrichtung der Testklasse, die dem von einem Argument gelieferten virtuellen Instrument zugeordnet ist, und das Leistungsspektrum und setzt dieses in Zuordnungsbeziehung der Blatttestklasse. Anschließend spezifiziert er die von einem Argument gelieferten Einsprungpunkte des virtuellen Instruments und der Leistungsspektrummessung selbst, um den Prozeß Initialisierung der Implementierungsvorrichtung der Testklasse auszuführen.

Setze Testbedingung (Einsprungpunkt des Testbedingungsdateinamens)

Der Prozeß Setze-Testbedingung liest eine Testbedingungsdatei im Speicher des Namens, der zu dem Attributnamen, des von einem Argument gelieferten Testbedingungsdateinamens paßt. Anschließend setzt er den gelesenen Inhalt in die Attribute Mittenfrequenz und Frequenzspanne ein.

Setze Schlitzinformation (Einsprungpunkt der Schlitzinformation)

Der Prozeß Setze-Schlitzinformation greift auf die von einem Argument gelieferte Schlitzinformation zu, um den Inhalt der Attribute Mittenfrequenz und Frequenzspanne zu setzen. Er greift auf die Attribute "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind" der von einem Argument gelieferten Schlitzinformation zu und setzt im Attribut Mittenfrequenz die Mittenfrequenzen der "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind". Er greift auf die Attribute "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind" der von einem Argument gelieferten Schlitzinformation zu und setzt im Attribut Frequenzspanne eine Differenzfrequenz zwischen den Frequenzen der "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind".

## Linienspektrummessung

[Stamm-Objekt]: Blatttestklasse

[Zuordnungsbeziehung]: keine

[Attribut]

Meßfrequenz

Dies ist ein Bereich zum Speichern der Meßfrequenz für die Messung eines Linienspektrums im Speicher 26.

[Prozeß]

Initialisierung (Einsprungpunkt des virtuellen Instruments)

Der Prozeß Initialisierung erzeugt eine Implementierungsvorrichtung der Testklasse, die dem virtuellen Instrument und dem Linienspektrum zugeordnet ist, die von einem Argument geliefert werden, und setzt diese in Zuordnungsbeziehung der Blatttestklasse. Anschließend spezifiziert er die Einsprungpunkte des virtuellen Instruments selbst und der Linienspektrummessung selbst, die von einem Argument geliefert werden, um den Prozeß Initialisierung der Implementierungsvorrichtung der Testklasse auszuführen.

Setze Testbedingung (Einsprungpunkt des Testbedingungsdateinamens)

Der Prozeß Setze-Testbedingung liest eine Testbedingungsdatei im Speicher 26 des Namens, der mit dem Attribut Name des von einem Argument gelieferten Testbedingungsdateinamens übereinstimmt. Anschließend setzt er den gelesenen Inhalt in das Attribut Meßfrequenz ein.

Setze Schlitzinformation (Einsprungpunkt der Schlitzinformation)

Der Prozeß Setze-Schlitzinformation greift auf die von einem Argument gelieferte Schlitzinformation zu, um den Inhalt der Meßfrequenz zu setzen. Er greift auf die Attribute "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind" der von einem Argument gelieferten Schlitzinformation zu und setzt im Attribut Meßfrequenz die Mittenfrequenzen der "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind" und "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind".

Zusammengesetzte Testklasse

[Stamm-Objekt]: Testklasse

[Zuordnungsbeziehung]

Meßklasse für Hochgeschwindigkeitsmessung

Dies ordnet eine Meßklasse für die Hochgeschwindigkeitsmessung zu.

Meßklasse für hochgenaue Messung

Dies ordnet mehrere Meßklassen für die hochgenaue Messung zu.

[Attribut]: keines

[Prozeß]: keiner

Oberwellenmessung

[Stamm-Objekt]: zusammengesetzte Testklasse

[Zuordnungsbeziehung]: keine

[Attribut]

Grundfrequenz

Dies ist ein Bereich zum Speichern der Grundfrequenz im Speicher 26.

Ordnung der Messung

Dies ist ein Bereich zum Speichern der Ordnung der zu messenden höchsten Oberwelle im Speicher 26.

[Prozeß]

Initialisierung (Einsprungpunkt des virtuellen Instruments) 25

Der Prozeß Initialisierung führt die folgenden Prozesse aus:

- 1) Erzeuge eine Leistungsspektrummessung für die Hochgeschwindigkeitsmessung und setze diese in die Messungsklasse für die Hochgeschwindigkeitsmessung der enthaltenen zusammengesetzten Testklasse ein.
- 2) Setze die Attribute Mittenfrequenz und Frequenzspanne der erzeugten Leistungsspektrummessung. Die Mittenfrequenz ist eine Frequenz zwischen der Grundwelle und den Oberwellen höchster Ordnung. Die Frequenzspanne ist eine Frequenz gleich oder größer als die Differenz zwischen der Grundwelle und der Oberwelle höchster Ordnung.
- 3) Spezifiziere einen Einsprungpunkt des von einem Argument gelieferten virtuellen Instruments, um den Prozeß Initialisierung der erzeugten Leistungsspektrummessung auszuführen.
- 4) Erzeuge die Linienspektrummessung für die hochgenaue Messung für die Anzahl der Ordnungen der Grundwelle und der zu messenden Oberwellen und setze diese ein in die Messungsklasse für hohe Genauigkeit der enthaltenen zusammengesetzten Testklasse.
- 5) Setze die Frequenzen der Grundwelle und der zu messenden Oberwellen im jeweiligen Attribut Meßfrequenz der erzeugten Linienspektrummessung.
- 6) Spezifiziere einen Einsprungpunkt des von einem Argument gelieferten virtuellen Instruments, um den Prozeß Initialisierung aller erzeugten Linienspektrummessungen auszuführen.

Setze Testbedingung (Einsprungpunkt des Testbedingungsdateinamens)

Der Prozeß Setze-Testbedingung liest eine Testbedingungsdatei im Speicher 26, deren Name mit dem Attributnamen des von einem Argument gelieferten Testbedingungsdateinamens übereinstimmt. Anschließend setzt er den gelesenen Inhalt in die Attribute Grundfrequenz und Meßordnung ein.

Nehme Testergebnis auf, das dem Testgegenstand zugeordnet ist (Einsprungpunkt-Satz der Testdaten, Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Prozeß "Nehme Testergebnis auf, das dem Testgegenstand zugeordnet ist" greift auf den Satz der Einsprungpunkte der von einem Argument gelieferten Testdaten zu, um ein Leistungsverhältnis zwischen der Grundwelle und den Oberwellen zu berechnen. Anschließend setzt er das berechnete Ergebnis in eine Zeichenkette um und setzt diese in den Bereich ein, der durch den Einsprungpunkt der Zeichenkette des Testergebnisses, das dem von einem Argument gelieferten Testgegenstand zugeordnet ist, angegeben wird.

Implementierungsvorrichtung der Testklasse

[Stammobjekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]

Virtuelles Instrument

Dies ist ein Bereich zum Speichern eines Einsprungpunktes des virtuellen Instruments, für das der Testparameter erzeugt worden ist.

Blatttestklasse

Dies ist ein Bereich zum Speichern eines Einsprungpunktes der Blatttestklasse, die die Implementierungsvorrichtung der Testklasse erzeugt.

[Prozeß]

Initialisierung (Einsprungpunkt des virtuellen Instruments, Einsprungpunkt der Blatttestklasse)

Der Prozeß Initialisierung speichert das von einem Argument gelieferte virtuelle Instrument im Attribut virtuelles Instrument. Außerdem speichert er die von einem Argument gelieferte Blatttestklasse im Attribut Blatttestklasse.

Erzeuge Testparameter (Einsprungpunkt des Testparameters)

Der Prozeß Erzeuge-Testparameter der Implementierungsvorrichtung der Testklasse bewirkt nichts. Dies ist ein Prozeß, der so vorbereitet ist, daß er den Prozeß Erzeuge-Testparameter für ein Abkömmling-Objekt ausführen kann, wenn das Abkömmling-Objekt als Implementierungsvorrichtung der Testklasse behandelt wird.

Implementierungsvorrichtung für Leistungsspektrummes-  
sung für virtuellen Spektrumanalysierer

[Stammobjekt]: Implementierungsvorrichtung der Test-  
klasse

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Erzeuge Testparameter (Einsprungpunkt des Testpara-  
meters)

Der Prozeß Erzeuge-Testparameter führt die folgenden  
Prozesse aus:

- 1) Zugreifen auf das Attribut virtuelles Instrument der  
enthaltenen Implementierungsvorrichtung der Test-  
klasse und Erhalten eines Einsprungpunktes des virtu-  
ellen Instruments.
- 2) Lesen des Attributs Mittenfrequenz der Leistungs-  
spektrummessung, das als Attribut Blatttestklasse der  
enthaltenen Implementierungsvorrichtung der Test-  
klasse gespeichert ist.
- 3) Spezifizieren des Einsprungpunktes des von einem  
Argument gelieferten Testparameters, des Wertes des  
Attributs Mittenfrequenz und des Namens der Mitten-  
frequenz und Ausführen des Prozesses Setze-Testpara-  
meter des erhaltenen virtuellen Instruments.
- 4) Lesen des Attributs Frequenzspanne der Leistungs-  
spektrummessung, das als Attribut Blatttestklasse der  
enthaltenen Implementierungsvorrichtung der Test-  
klasse gespeichert ist.
- 5) Spezifizieren des Einsprungpunktes des von einem  
Argument gelieferten Testparameters, des Wertes des  
Attributs Frequenzspanne und den Namen der Fre-  
quenzspanne und Ausführen des Prozesses Setze-Test-  
parameter des erhaltenen virtuellen Instruments.

Implementierungsvorrichtung für Linienspektrummessung  
für virtuellen Spektrumanalysator

[Stammobjekt]: keines

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Erzeuge Testparameter Einsprungpunkt des Testparameters

Der Prozeß Erzeuge-Testparameter führt die folgenden  
Prozesse aus:

- 1) Zugreifen auf das Attribut virtuelles Instrument der  
enthaltenen Implementierungsvorrichtung der Test-  
klasse und Erhalten eines Einsprungpunktes des virtu-  
ellen Instruments.
- 2) Lesen des Attributs Meßfrequenz der Linienspek-  
trummessung, das als Attribut Blatttestklasse der  
enthaltenen Implementierungsvorrichtung der Testklasse  
gespeichert ist.
- 3) Spezifizieren des Einsprungpunktes des von einem  
Argument gelieferten Testparameters, des Wertes des

Attribut Meßfrequenz und des Namens der Mittenfre-  
quenz und Ausführen des Prozesses Setze-Testpara-  
meter des erhaltenen virtuellen Instruments.

Schlitzinformation

[Stammobjekt]: keines

[Zuordnungsbeziehung]

Testdaten

Die Schlitzinformation ordnet die Testdaten zu, die die  
Basis der erzeugten graphischen Darstellung sind.

[Attribut]

X-Achsen-Daten, die dem Minimalwert auf der X-Achse  
zugeordnet sind

Dies ist ein Bereich zum Speichern der X-Achsen-Daten,  
die dem Minimalwert auf der X-Achse zugeordnet sind.

X-Achsen-Daten, die dem Maximalwert auf der X-Achse  
zugeordnet sind

Dies ist ein Bereich zum Speichern der X-Achsen-Daten,  
die dem Maximalwert auf der X-Achse zugeordnet sind.

Y-Achsen-Daten, die dem Minimalwert auf der Y-Achse zu-  
geordnet sind

Dies ist ein Bereich zum Speichern der Y-Achsen-Daten,  
die dem Minimalwert auf der Y-Achse zugeordnet sind.

Y-Achsen-Daten, die dem Maximalwert auf der Y-Achse zu-  
geordnet sind

Dies ist ein Bereich zum Speichern der Y-Achsen-Daten,  
die dem Maximalwert auf der Y-Achse zugeordnet sind.

Spezifizierter virtueller Instrumentendienst

Dies ist ein Bereich zum Speichern der Matrixdaten der  
Einsprungpunkte des virtuellen Instrumentendienstes.

Spezifiziertes virtuelles Instrument

Dies ist ein Bereich zum Speichern der Matrixdaten der  
Einsprungpunkte des virtuellen Instruments.

Darstellungsblatt

Dies ist ein Bereich zum Zuordnen des Einsprungpunktes  
des von der Schlitzinformation erzeugten Darstellungsblat-  
tes.

[Prozeß]

Initialisierung (X-Achsen-Daten, die dem Minimalwert auf  
der X-Achse zugeordnet sind, X-Achsen-Daten, die dem  
Maximalwert auf der X-Achse zugeordnet sind, Y-Achsen-  
Daten, die dem Minimalwert auf der Y-Achse zugeordnet  
sind, Y-Achsen-Daten, die dem Maximalwert auf der Y-  
Achse zugeordnet sind)

Dieser Prozeß speichert die X-Achsen-Daten, die dem



Minimalwert auf der X-Achse zugeordnet sind, X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind, Y-Achsen-Daten, die dem Minimalwert auf der Y-Achse zugeordnet sind, Y-Achsen-Daten, die dem Maximalwert auf der Y-Achse zugeordnet sind, die von einem Argument geliefert werden, in den Attributen "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind", "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind", "Y-Achsen-Daten, die dem Minimalwert auf der Y-Achse zugeordnet sind", "Y-Achsen-Daten, die dem Maximalwert auf der Y-Achse zugeordnet sind".

#### Aktualisierung ()

Dies zeigt die äußeren Rahmen der Bereiche an, die durch die Werte der Attribute "X-Achsen-Daten, die dem Minimalwert auf der X-Achse zugeordnet sind", "X-Achsen-Daten, die dem Maximalwert auf der X-Achse zugeordnet sind", "Y-Achsen-Daten, die dem Minimalwert auf der Y-Achse zugeordnet sind", "Y-Achsen-Daten, die dem Maximalwert auf der Y-Achse zugeordnet sind" angegeben werden, auf der graphischen Darstellung des vom Attribut Darstellungsbereich angegebenen Darstellungsbereiches.

Spezifiziere Testelementnamen (Einsprungpunkt des Testelementnamens)

Der Prozeß Spezifiziere-Testelementnamen zeigt zuerst das Graphiksymbol des von einem Argument gelieferten Testelementnamens auf der graphischen Darstellung an. Anschließend sucht er eine Wechselwirkungsrichtung, spezifiziert die Einsprungpunkte des von einem Argument gelieferten Testelementnamens und der Schlitzinformation selbst und führt den Prozeß Starte-Neuen-Test der Wechselwirkungsrichtung aus.

Spezifiziere virtuellen Instrumentendienst (Einsprungpunkt des virtuellen Instrumentendienstes)

Der Prozeß Spezifiziere-Virtuellen-Instrumentendienst spezifiziert den Einsprungpunkt der Schlitzinformation selbst, um den Prozeß Dienstaussführung des von einem Argument gelieferten virtuellen Instrumentendienstes auszuführen.

Wechselwirkungsrichtung

[Stammobjekt]: keines

[Zuordnungsbeziehung]

Virtuelles Instrument

Dies ordnet mehrere virtuelle Instrumente zu, die vom Testsystem verwendet werden können.

[Attribut]: keines

[Prozeß]

Starte neuen Test (Einsprungpunkt des Testelementnamens, Einsprungpunkt der Schlitzinformation)

Der Prozeß Starte-Neuen-Test greift auf den durch Vorgabe gestarteten Testgegenstand des von einem Argument gelieferten Testelementnamens zu und führt die folgenden Prozesse aus:

(a) Wenn ein durch Vorgabe gestarteter Testgegenstand gespeichert ist

Dies spezifiziert die Einsprungpunkte des durch Vorgabe gestarteten Testgegenstandes, des von einem Argument gelieferten Testelementnamens und der Schlitzinformation, um den Prozeß Messungsausführung auszuführen.

(b) Wenn ein durch Vorgabe gestarteter Testgegenstand nicht gespeichert ist

1) Dies spezifiziert einen graphischen Darstellungsbereich, zeigt das Graphiksymbol des Testgegenstandes, dem der von einem Argument gelieferte Testelementname zugeordnet ist, auf der graphischen Darstellung an und wartet, bis das Graphiksymbol durch Zeigen mit der Zeigevorrichtung 30 ausgewählt ist (Drücken des Druckknopfes der Zeigevorrichtung 30 und Loslassen des Druckknopfes der Zeigevorrichtung 30 ohne Bewegen der Zeigevorrichtung)

2) Sobald das Graphiksymbol des Testgegenstandes ausgewählt ist, spezifiziert dies die Einsprungpunkte des Testgegenstandes, der dem ausgewählten Graphiksymbol zugeordnet ist, des von einem Argument gelieferten Testelementnamens und der Schlitzinformation und führt den Prozeß Messungsausführung aus.

Messungsausführung (Einsprungpunkt des Testgegenstandes, Einsprungpunkt der Schlitzinformation, Einsprungpunkt des Testelementnamens)

Der Prozeß Messungsausführung führt die folgenden Prozesse aus.

1) Erfasse ein virtuelles Instrument, das dem von einem Argument gelieferten Testelementnamen zugeordnet ist.

2) Spezifiziere die Einsprungpunkte des von einem Argument gelieferten Testgegenstandes und der Schlitzinformation, um den Prozeß "erzeuge Testparameter mit Testgegenstand und Schlitzinformation" des erfaßten virtuellen Instruments auszuführen.

3) Spezifiziere einen Einsprungpunkt des erzeugten Testparameters und führe den Prozeß Testausführung des erfaßten virtuellen Instruments aus.

Aufgabenkomponente

[Stammobjekt]: keines

[Attribut]

Name

Dies ist ein Bereich zum Speichern einer Zeichenkette, die einen Namen darstellt, im Speicher 26.

Anzeigebild

Dies ist ein Bereich zum Speichern eines Anzeigebildes eines Graphiksymbols im Speicher 26.

Größe des Graphiksymbols in X-Achsen-Richtung

Dies ist ein Bereich zum Speichern der Größe des angezeigten Graphiksymbols in X-Achsen-Richtung.

## Größe des Graphiksymbols in Y-Achsen-Richtung

Dies ist ein Bereich zum Speichern der Größe des angezeigten Graphiksymbols in Y-Achsen-Richtung.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Prozeß Ausführung der Aufgabenkomponente bewirkt nichts. Er ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Ausführung für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt wie eine Aufgabenkomponente behandelt wird.

Anbringung (Einsprungpunkt der Aufgabenkomponente, Koordinatenposition)

Der Prozeß Anbringung ist ein Prozeß, der ausgeführt wird, wenn das Graphiksymbol eines Programmelements auf das Graphiksymbol der Aufgabenkomponente fallengelassen wird. Der Prozeß Anbringung der Aufgabenkomponente bewirkt nichts. Er ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Anbringung für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt wie eine Aufgabenkomponente behandelt wird.

Verbinde ()

Der Prozeß Verbinde der Aufgabenkomponente bewirkt nichts. Er ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Verbinde für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt wie eine Aufgabenkomponente behandelt wird.

Löse Verbindung ()

Der Prozeß Löse-Verbindung der Aufgabenkomponente bewirkt nichts. Er ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Löse-Verbindung für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt wie eine Aufgabenkomponente behandelt wird.

Ausrichtung

Der Prozeß Ausrichtung der Aufgabenkomponente bewirkt nichts. Er ist ein Prozeß, der so vorbereitet ist, daß der Prozeß Ausrichtung der Aufgabenkomponente für ein Abkömmling-Objekt selbst dann ausgeführt werden kann, wenn das Abkömmling-Objekt wie eine Aufgabenkomponente behandelt wird.

Erzeuge Graphiksymbol (Koordinatenposition, Anzeigebereich)

Er greift auf das Attribut Anzeigebild zu, um das Graphiksymbol an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich anzuzeigen.

Nehme Graphiksymbolgröße auf (Einsprungpunkt der Größendaten in X-Achsen-Richtung, Einsprungpunkt der Größendaten in Y-Achsen-Richtung)

Der Prozeß "nehme Graphiksymbolgröße auf" speichert zuerst die Größe des anzuzeigenden Graphiksymbols in X-

Achsen-Richtung in einem Bereich, der durch den Einsprungpunkt der X-Achsen-Richtung-Größendaten eines Arguments angegeben wird. Anschließend speichert er die Größe des anzuzeigenden Graphiksymbols in Y-Achsen-Richtung in einem Bereich, der durch den Einsprungpunkt der Y-Achsen-Richtung-Größendaten eines Arguments angegeben wird.

Beginn

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Der Prozeßausführung des Beginns ist ein Prozeß, der ausgeführt wird, wenn auf das Graphiksymbol Beginn gezeigt wird. Er erfaßt einen Testplan, der den Beginn zuordnet, und führt den Prozeß Ausführung des erfaßten Testplans aus.

Ende

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]: keiner

Pause

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]: keine

[Attribut]: keines

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dies stoppt das Programm, bis eine Eingabe von der Zeitvorrichtung 30 vorliegt.

Warten

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]: keine

[Attribut]

Wartezeit

Dies ist ein Bereich zum Speichern einer Zeitperiode, die ein Programm wartet.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dieser wartet für ein Programm für eine im Attribut Wartezeit spezifizierte Zeitperiode.

Warte bis zu spezifisiertem Zeitpunkt

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]: keine

[Attribut]

Startzeitpunkt

Dies ist ein Bereich zum Speichern des Zeitpunktes zum Starten des Programms.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dies ist ein Prozeß zum Anhalten des Programms, bis der Startzeitpunkt erreicht ist.

Zusammengesetzte Aufgabe

[Stammobjekt]: Aufgabenkomponente

[Zuordnungsbeziehung]

Aufgabenkomponente

Dies ordnet mehrere Aufgabenkomponenten zu, die das Programm bilden.

[Attribut]: keines

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dies führt die Prozesse "Ausführung der zugeordneten Aufgabenkomponenten" von der Spitze der Matrixdaten aus, die die Zuordnungsbeziehung speichern. Er führt jedoch nicht den Prozeßbeginn aus.

Anbringung (Einsprungpunkt der Aufgabenkomponente, Koordinatenposition)

Der Anbringungsprozeß führt die folgenden Prozesse aus:

- 1) Ausführung des Prozesses Lösche-Verbindung.
- 2) Erzeuge eine Kopie der von einem Argument gelieferten Aufgabenkomponente.
- 3) Spezifiziere die von einem Argument gelieferten Positionskordinaten und den Graphiksymbolbereich der zusammengesetzten Aufgabe und führe den Prozeß Erzeuge-Graphiksymbol der Aufgabenkomponente aus.
- 4) Führe den Prozeß Ausrichtung aus.
- 5) Führe den Prozeß Verbinde aus.
- 6) Füge die erzeugten Aufgabenkomponenten hinzu,

so daß sie der Anzeigereihenfolge der Programmementographiksymbole zwischen den angezeigten Symbolen Beginn und Ende entsprechen, und richte die Matrixdaten der Einsprungpunkte der Programmelemente, die die zusammengesetzte Aufgabe speichert, als Zuordnungsbeziehung erneut aus.

Verbinde ()

Dies verbindet die benachbarten Graphiksymbole von Beginn bis Ende und zeigt diese an.

Lösche Verbindung ()

Dies löscht die Verbindungsanzeige von Beginn bis Ende.

Ausrichtung ()

Dies verbindet zuerst die Größe der Graphiksymbole von Beginn bis Ende und zeigt diese an durch Verwendung des Prozesses "Nehme Größe des Graphiksymbols auf" und modifiziert das Graphiksymbol entsprechend der Größe der zusammengesetzten Aufgabe selbst zu einer Größe, die alle Graphiksymbole anzeigen kann. Anschließend richtet er die Graphiksymbole von Beginn bis Ende aus, so daß sie verbunden und angezeigt werden können. Wenn eine zusammengesetzte Aufgabe vorliegt, die die zusammengesetzte Aufgabe selbst zuordnet, führt er schließlich den Prozeß Ausrichtung für das Zuordnen der zusammengesetzten Aufgabe aus.

Schleife

[Stammobjekt]: zusammengesetzte Aufgabe

[Zuordnungsbeziehung]: keine

[Attribut]

Anzahl der Schleifen

Dies ist ein Bereich zum Speichern der Anzahl der Schleifen.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dies führt den Prozeß Ausführung der enthaltenen zusammengesetzten Aufgabe in der durch Schleifenanzahl spezifizierten Anzahl aus.

Erzeuge Graphiksymbol (Koordinatenposition, Anzeigebereich)

Dies zeigt die Graphiksymbole mit einem Schleifenbild und einer Eingabetafel an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich an.

[Stammobjekt]: zusammengesetzte Aufgabe

[Zuordnungsbeziehung]: keine

[Attribut]

Ausführungsbedingung

Dies ist ein Bereich zum Speichern der Ausführungsbedingung als Zeichenkette.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Das Attribut Ausführungsbedingung umfaßt eine Zeichenkette, die aus numerischen Werten, dem Gleichheitszeichen und einem negativen Vorzeichen besteht. Der Prozeß Ausführung ermittelt zuerst entsprechend dem Attribut Ausführungsbedingung, ob ein numerischer Wert, der einer Zeichenkette des Testergebnisses, das einem von einem Argument gelieferten Testgegenstand entspricht, z. B. größer ist als ein bestimmter numerischer Wert, oder innerhalb eines Bereiches zwischen einem oberen Grenzwert und einem unteren Grenzwert liegt. Anschließend führt der Prozeß Ausführung, wenn die vom Attribut Ausführungsbedingung gelieferte Bedingung erfüllt ist, den Prozeß Ausführung der enthaltenen zusammengesetzten Aufgabe aus.

Erzeuge Graphiksymbol (Koordinatenposition, Anzeigebereich)

Dies zeigt die Graphiksymbole mit einem Verzweigungsbild und einer Eingabetafel an den von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich an.

Bedingte Schleife

[Stammobjekt]: zusammengesetzte Aufgaben

[Zuordnungsbeziehung]: keine

[Attribut]

Abschlußbedingung

Dies ist ein Bereich zum Speichern einer Bedingung, die eine Schleife beendet, als Zeichenkette.

[Prozeß]

Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Die Abschlußbedingung umfaßt eine Zeichenkette, die numerische Werte, ein Gleichheitszeichen und ein negatives Vorzeichen umfaßt. Die Einzelheiten der Verarbeitung des Prozesses Ausführung sind folgende.

- 1) Führe den Prozeß Ausführung der enthaltenen zusammengesetzten Aufgabe aus, um eine Zeichenkette des Testergebnisses zu erhalten, das dem Testgegenstand zugeordnet ist, und liefere diese als Ergebnis des Prozesses.

- 2) Führe anschließend die vom Attribut Abschlußbedingung gelieferte Ermittlung durch, ob ein numerischer Wert, der der erhaltenen Zeichenkette entspricht, z. B. größer ist als ein bestimmter numerischer Wert oder innerhalb eines Bereiches zwischen einem oberen Grenzwert und einem unteren Grenzwert liegt.

- 3) Schließe den Prozeß ab, wenn der Prozeß Ausführung eine vom Attribut Abschlußbedingung gelieferte Bedingung erfüllt. Wenn die Bedingung nicht erfüllt ist, wiederhole den Prozeß ausgehend von (1).

Erzeuge Graphiksymbol (Koordinatenposition, Anzeigebereich)

Dies zeigt die Graphiksymbole mit einem Verzweigungsbild und einer Eingabetafel an dem von einem Argument gelieferten Positionskordinaten auf dem von einem Argument gelieferten Anzeigebereich an.

Testplan

[Stammobjekt]: zusammengesetzte Aufgabe

[Zuordnungsbeziehung]

Zusammengesetzte Aufgabe

Dies ordnet mehrere Aufgabenkomponenten zu, die verfügbare Programmelemente sind.

[Attribut]

Basisentwurf des Testplans

Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf der graphischen Anzeigevorrichtung, den der Testplan verwendet, im Speicher 26.

Programmelementanzeige-Entwurf

Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Basisentwurf des Testplans zum Anzeigen eines verfügbaren Programmelements im Speicher.

Testplancntwurf

Dies ist ein Bereich zum Speichern eines Anzeigebereichs auf dem Basisentwurf des Testplans zum Erzeugen eines Programms im Speicher.

[Prozeß]

Initialisiere Testplan ()

Der Prozeß Initialisiere-Testplan führt die folgenden Prozesse aus:

- 1) Führe den Prozeß Erzeuge-Anzeigebereich aus.
- 2) Erzeuge ein für ein Programm verfügbares Programmelement und setze dieses in Zuordnungsbeziehung der zusammengesetzten Aufgabe des Testplans. Zeige anschließend das verfügbare Programmelement auf einem Bereich an, der vom Attribut Programmelementanzeigeentwurf angegeben wird.
- 3) Erzeuge Beginn und Ende und setze diese in Zuordnungsbeziehung der enthaltenen zusammengesetzten Aufgabe. Erzeuge anschließend die Graphiksym-

bole von Beginn und Ende und zeige diese auf dem Testplanentwurf an.

- 4) Führe den Prozeß Ausrichtung aus.
- 5) Führe den Prozeß Verbinde aus.

#### Erzeuge Anzeigebereich ()

Dies erzeugt Anzeigebereiche, die den Attributen Basisentwurf des Testplans, Programmelementanzeigeentwurf und Testplanentwurf zugeordnet sind, auf der graphischen Anzeigevorrichtung und speichert diese im jeweiligen Attribut.

#### Zeige verfügbares Programmelement an (Anzeigebereich)

Dies erzeugt ein Graphiksymbol des zugeordneten Testbedingungsdateinamens auf dem von einem Argument gelieferten Anzeigebereich.

#### Ausführung (Einsprungpunkt der Zeichenkette des Testergebnisses, das dem Testgegenstand zugeordnet ist)

Dies führt den Prozeß Ausführung der enthaltenen zusammengesetzten Aufgabe aus.

Im folgenden werden verschiedene Prozeßbeispiele beschrieben, die unter Verwendung des jeweils obenbeschriebenen Objektes implementiert werden. Mit dieser Verbindung umreißt das folgende eine Beschreibung des Testzieles und des Prozesses zum Starten eines Tests, wenn das jeweilige obenbeschriebene Objekt verwendet wird. Da außerdem die Einzelheiten des jeweiligen Prozesses für den Prozeß jedes Objektes beschrieben werden, zeigt das folgende, wie das jeweilige Objekt gestartet wird.

(1) Prozeßbeispiel des Testsystems, das spezifizieren kann, ob der Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit durchgeführt wird, und daß die Operation starten kann durch Auswählen des Test-Zwecks/Ziels

#### (1.0) Implementierungsbeispiel des GUI-Testsystems, das die Operation starten kann durch Auswählen eines Test-Zwecks/Ziels

Die Fig. 14 zeigt ein Beispiel der GUI-Implementierung in einem Testsystem, das die Operation starten kann durch Auswählen eines Test-Zwecks/Ziels. Wie in der Figur gezeigt, enthält ein Basisentwurf der Testaufgabe 300 einen Testelementanzeigeentwurf 310, einen Testdatenanzeigeentwurf 320 und einen Testaufgabentwurf 330.

Der Basisentwurf 300 ist ein Teil des Anzeigebereichs oder der gesamte Anzeigebereich der graphischen Anzeigevorrichtung, der zum Setzen eines Test-Zwecks/Ziels verwendet wird, um den Test auszuführen und das Testergebnis anzuzeigen. Außerdem ist der Testelementanzeigeentwurf 310 ein Anzeigebereich auf dem Basisentwurf der Testaufgabe 300 zum Anzeigen des Testelements, das im Testsystem dieser Ausführungsform verwendet werden kann.

Der Testdatenanzeigeentwurf 320 ist ein Anzeigebereich auf dem Basisentwurf der Testaufgabe 300 zum Anzeigen der Testdaten. Außerdem ist ein Bereich 322 zum Anzeigen der Testdaten im Testdatenanzeigeentwurf 320 angeordnet.

Der Testaufgabentwurf 330 ist ein Anzeigebereich auf dem Basisentwurf der Testaufgabe 300 zum Spezifizieren eines Testziels und zum Starten des Tests. Dieser Testaufgabentwurf 330 umfaßt einen Testgegenstandentwurf 332 zum Anzeigen eines Testgegenstandes, einen Testbedingungsentwurf 336 zum Anzeigen eines Testbedingungsda-

teinamens, einen Arbeitsraum 340 zum Anzeigen einer DUT, des ausgewählten Testgegenstandes und der Testbedingung, die Testelemente, die dem Hochgeschwindigkeitstest oder dem hochgenauen Test zugeordnet sind oder dergleichen und verwendet werden zum Verbinden zwischen den Graphiksymbolen der DUT, den Testelementen, die dem Hochgeschwindigkeitstest oder dem hochgenauen Test zugeordnet sind, oder dergleichen, sowie ein Graphiksymbol der Testaufgabe 350 zum Auswählen einer Testaufgabe als Programmelement.

Jeder dieser Anzeigebereiche wird erzeugt vom Prozeß Erzeuge-Anzeigebereich der Testaufgabe. Anschließend werden die Graphiksymbole 312 und 314 des Testelementnamens, der die Testelemente im Testelementanzeigeentwurf 310 darstellt, ein Graphiksymbol der DUT 342 im Arbeitsraum 340 erzeugt, wobei das System auf die Betätigung durch den Benutzer des Testsystems wartet.

Wenn anschließend der Benutzer des Testsystems ein Graphiksymbol des Testelementnamens, der ein Testelement darstellt, aus dem Testelementanzeigeentwurf 310 herauszieht und auf den Arbeitsraum 340 fallen läßt, werden die Graphiksymbole 344, 346 der Testelementnamen, die dem Hochgeschwindigkeitstest und dem hochgenauen Test zugeordnet sind, im Arbeitsraum 340 angezeigt, während ein Graphiksymbol 334 des Testgegenstandes, der für jedes Testelement verfügbar ist, auf dem Testgegenstandentwurf 332 angezeigt wird.

Wenn in einem solchen Zustand das Graphiksymbol des Testgegenstandes 334 auf dem Testgegenstandentwurf 310 gezogen und auf den Arbeitsraum 340 fallengelassen wird, wird das Graphiksymbol des Testgegenstandes 334 gelöscht und ein Graphiksymbol des Testgegenstandes 348 auf dem Arbeitsraum 340 neu angezeigt. Gleichzeitig wird ein Graphiksymbol 338 des Testbedingungsdateinamens oder dergleichen, der für den Testgegenstand verfügbar ist, auf dem Testbedingungsentwurf 336 angezeigt.

Wenn das Graphiksymbol des Testbedingungsdateinamens 338, das auf dem Testbedingungsentwurf 336 angezeigt wird, gezogen und auf dem Arbeitsraum 340 fallengelassen wird, wird das Graphiksymbol des Testbedingungsdateinamens 338 gelöscht, während ein Graphiksymbol des Testbedingungsdateinamens 339 auf dem Arbeitsraum 340 erzeugt wird.

Der Testgegenstandentwurf 332 und der Testbedingungsentwurf 336 besitzen viele Graphiksymbole, wobei dann, wenn diese nicht in diesen Anzeigebereichen aufgenommen werden können, ein sogenannter Rollkahn für einen Teil von diesen angezeigt wird, der ein Rollen erlaubt.

Wenn das DUT-Graphiksymbol 342 gezogen und auf einem der Graphiksymbole 344 oder 346, die dem Hochgeschwindigkeitstest oder dem hochgenauen Test zugeordnet sind, fallengelassen wird, wird das DUT-Graphiksymbol 342 mit dem fallengelassenen Graphiksymbol verbunden und angezeigt, so daß der Hochgeschwindigkeitstest oder der hochgenaue Test ausgeführt wird. Das heißt, eine Ziehen-und-Fallengelassen-Operation des DUT-Graphiksymbols 342 ermöglicht, den Hochgeschwindigkeitstest oder den hochgenauen Test auszuwählen und die Ausführung des Tests auszuweisen.

Wenn der Test ausgeführt wird, wird der Testdatenangeentwurf 320 angezeigt und ein Graph in einer Stufe angezeigt, in der die Testdaten aufgenommen werden.

#### (1.1) Prozeß zum Spezifizieren des Testziels und zum Starten des Tests

Die Fig. 15 ist ein Flußdiagramm, das eine Operationsprozedur beim Starten eines Tests durch Spezifizieren eines

Testziels zeigt. Zuerst zeigt der Initialisierungsprozeß der Testaufgabe (Schritt d1) den Basisentwurf der Testaufgabe **300**, den Testelementanzengeentwurf **310**, den Testgegenstandsentwurf **332**, den Testbedingungsentwurf **336** und den in **Fig. 13** gezeigten Arbeitsraum **340** auf der graphischen Anzeigevorrichtung an und zeigt ferner ein Graphiksymbol des verfügbaren Testelementnamens auf dem Testelementanzengeentwurf **310** an.

Wenn in einem solchen Zustand das Graphiksymbol des Testelementnamens auf den Testgegenstandsentwurf **310** gezogen und auf den Arbeitsraum **340** fallengelassen wird, wird ein Graphiksymbol des Testelementnamens, der dem Hochgeschwindigkeitstest oder dem hochgenauen Test Zugeordnet ist, auf dem Arbeitsraum **340** angezeigt, während ein Graphiksymbol des für das Testelement verfügbaren Testgegenstandes auf dem Testgegenstandsentwurf **332** angezeigt wird (Schritt d2).

Anschließend wird die Operation durchgeführt, die in Abhängigkeit davon unterteilt wird, ob der Benutzer des Testsystems einen Testgegenstand auswählt (Schritt d3).

Wenn kein Testgegenstand ausgewählt ist, fährt die Operation mit Schritt d7 fort, der später beschrieben wird. Wenn ein Testgegenstand ausgewählt ist, d. h. wenn ein Graphiksymbol des Testgegenstandes auf den Testgegenstandsentwurf **332** gezogen und auf den Arbeitsraum **340** fallengelassen wird, setzt außerdem die Testaufgabe den ausgewählten Testgegenstand auf das Testziel (Schritt d4), löscht das Graphiksymbol des ausgewählten Testgegenstandes auf dem Testgegenstandsentwurf **332** und zeigt das Graphiksymbol des ausgewählten Testgegenstandes auf dem Arbeitsraum **340** an. Gleichzeitig wird ein Graphiksymbol des verfügbaren Testbedingungsdateinamens, der für den Testgegenstand verfügbar ist, auf dem Testbedingungsentwurf **336** angezeigt.

Wenn anschließend der Benutzer des Testsystems ein Graphiksymbol des Testbedingungsdateinamens auf den Testbedingungsentwurf **336** zieht und auf den Arbeitsraum **340** fallen läßt, um den Testbedingungsdateinamen auszuwählen (Schritt d5), setzt die Testaufgabe den ausgewählten Testbedingungsdateinamen auf das Testziel (Schritt d6), löscht das Graphiksymbol des ausgewählten Testbedingungsdateinamens auf dem Testbedingungsentwurf **336** und zeigt das Graphiksymbol des ausgewählten Testbedingungsdateinamens auf dem Arbeitsraum **340** an.

Wenn der Benutzer weder einen Testgegenstand noch einen Testbedingungsdateinamen auswählt und ein DUT-Graphiksymbol gezogen und auf den Arbeitsraum **340** fallengelassen wird (Schritt d7), wird der Hochgeschwindigkeitstest auf den Testkompromiß gesetzt, den das Testziel besitzt, wenn ein Graphiksymbol des Testelementnamens, das dem Hochgeschwindigkeitstest zugeordnet ist, an der Position vorhanden ist, an der das DUT-Graphiksymbol fallengelassen wird. Wenn ein Graphiksymbol des Testelementnamens, das dem hochgenauen Test zugeordnet ist, an einer Position vorhanden ist, an der das DUT-Graphiksymbol fallengelassen wird, wird außerdem der hochgenaue Test auf den Testkompromiß gesetzt, den das Testziel besitzt.

Anschließend wird das DUT-Graphiksymbol mit dem Testelementnamen verbunden, der an der Position vorhanden ist, an der das DUT-Graphiksymbol fallengelassen wird (Schritt d8), wobei der Ausführungsprozeß eines verfügbaren Tests gestartet wird (Schritt d9).

(1.2) Ausrichtung für Anzeige der Testdaten durch Ausführung des Tests

Die **Fig. 16** und **17** sind ein Flußdiagramm, das eine Operationsprozedur zeigt, wenn Testdaten als Ergebnis nach der

Durchführung eines Tests angezeigt werden. Eine Serie von Operationen, die in diesen Figuren gezeigt sind, wird als Antwort auf die Ausführung des Tests in einem Schritt d9 der **Fig. 15** ausgeführt.

Wenn der Ausführungsprozeß des Tests im Schritt d9 der **Fig. 15** gestartet wird, führt die Testaufgabe den Prozeß Starte-Test des virtuellen Instruments aus (Schritt e1) Die Operation jedes Schritts nach diesem entspricht dem Prozeß Starte-Test des virtuellen Instruments. Außerdem wird der Prozeß Starte-Test des virtuellen Instruments mit dem bereits erzeugten Testziel als Argument gesetzt.

Das virtuelle Instrument erzeugt zuerst einen Testparameter (Schritt e2) und setzt einen Testparametergegenstand im erzeugten Testparameter. Außerdem wird auf den Inhalt des Testziels zugegriffen und die entsprechende Testklasse erzeugt (Schritt e3).

Anschließend führt die erzeugte Testklasse den Prozeß Initialisierung aus (Schritt e4). Danach ermittelt die Testaufgabe, ob der Testbedingungsdateiname im Testziel gesetzt ist (Schritt e5), wobei dann, wenn er gesetzt ist, die Testklasse den Prozeß Setze-Testbedingung ausführt (Schritt e6), um die vom Testbedingungsdateinamen angegebene Testbedingung auf die Testklasse zu setzen.

Außerdem führt der Prozeß Starte-Test des virtuellen Instruments den Prozeß Erzeuge-Testparameter der erzeugten Testklasse aus, um den Inhalt des erzeugten Testparameters zu setzen (Schritt e7). Dieser Prozeß Erzeuge-Testparameter der Testklasse verwendet den Prozeß Setze-Testparametergegenstand des virtuellen Instruments, um den erzeugten Testparameter im Inhalt des Testparameters zu setzen (Schritt e8). Wenn der Testparameter vorbereitet ist, erzeugt der Prozeß Starte-Test des virtuellen Instruments die Testdaten zum Speichern der vom Testelement erhaltenen Daten (Schritt e9) und erzeugt ein Darstellungsblatt zum Anzeigen der Testdaten (Schritt e10).

Anschließend führt der Prozeß Starte-Test des virtuellen Instruments den Prozeß Testausführung des virtuellen Instruments aus (Schritt e11). Der Prozeß Testausführung des virtuellen Instruments sendet den erzeugten Testparameter zum Testelement, steuert das Testelement, um die vorgegebene Messung durchzuführen, und nimmt die Testdaten vom Testelement auf. Anschließend setzt der Prozeß Testausführung des virtuellen Instruments die aufgenommenen Daten in die Testdaten ein und führt den Prozeß Aktualisierung des Darstellungsblattes aus, um den Inhalt der Testdaten anzuzeigen (Schritt d12).

Wenn die Anzeige der Testdaten abgeschlossen ist, führt der Prozeß Starte-Test des virtuellen Instruments den Prozeß "Nehme Testergebnis auf, das dem Testgegenstand zugeordnet ist" der erzeugten Testklasse aus (Schritt e13) und erhält das Testergebnis, das dem Testgegenstand zugeordnet ist.

(2) Prozeßbeispiel der Funktion Starte-Dienst, die einen Bereich aufnimmt, der durch die Koordinaten auf der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses spezifiziert ist

## (2.0) Beispiel der GUI-Implementierung

Die **Fig. 18** ist ein Beispiel einer GUI-Implementierung in einem Testsystem, das einen entsprechenden Dienst startet, wenn bestimmte Koordinaten auf einer graphischen Darstellung eines Testergebnisses angegeben werden.

Wenn der Test ausgeführt worden ist, wird der Testdaten-anzeigenentwurf **320** angezeigt, wobei in der Stufe, in der die Testdaten aufgenommen werden, ein Graph in einem Bereich **322** zum Anzeigen eines Teils der Testdaten angezeigt wird.

Wenn in einem solchen Zustand der Benutzer einen Bereich 323 eines Teils des Graphen spezifiziert, wird ein Dienstausswahlentwurf 370 angezeigt, der ein Bereich zum Anzeigen des Graphiksymbols eines verfügbaren virtuellen Instruments ist, sowie ein Testelementauswahlentwurf 360, der ein Bereich ist zum Anzeigen verfügbarer Testelemente auf dem Testdatenanzeigeentwurf 320. Anschließend werden die Graphiksymbole 362 und 364 der Testelementnamen auf dem Testelementauswahlentwurf 360 angezeigt. Außerdem werden die Graphiksymbole der verfügbaren virtuellen Instrumentendienste auf dem Dienstausswahlentwurf 370 angezeigt. Die Fig. 18 zeigt ein Beispiel, bei dem ein Graphiksymbol 372 der Spitzenwerte, die ein vom virtuellen Instrumentendienst abgeleitetes Objekt ist, angezeigt wird.

Wenn das Graphiksymbol 372 der Spitzenwerte des Dienstausswahlentwurfs 370 gezogen und auf dem Bereich 323 eines Teils des Graphen fallengelassen wird, wird ein Graphiksymbol 324 angezeigt, das einen Spitzenwert entsprechend einem Spitzenwert im Bereich 323 anzeigt, wobei gleichzeitig Werte 325 angezeigt werden, die die Werte auf den X- und Y-Achsen sind.

Wenn somit Koordinaten auf der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses spezifiziert werden, ist es möglich, einen Dienst zu starten, den den spezifizierten Bereich aufnimmt.

(2.1) Ausführung des Dienstes zum Auffinden des Spitzenwertes in einem spezifizierten Bereich der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses

Die Fig. 19 ist ein Flußdiagramm, das eine Operationsprozedur zum Ausführen eines Dienstes zum Erhalten eines Spitzenwertes in einem spezifizierten Bereich zeigt.

Der Prozeß Hervorhebung des Darstellungsblattes wird ausgeführt, wenn ein Bereich auf der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses spezifiziert wird (Schritt f1). Der Prozeß Hervorhebung des Darstellungsblattes erzeugt zuerst die Schlitzinformation (Schritt f2) und führt den Prozeß Aktualisierung der Schlitzinformation aus (Schritt f3). Der Prozeß Aktualisierung der Schlitzinformation erzeugt einen äußeren Rahmen des Bereichs 323 eines Teils des Graphen auf der graphischen Darstellung.

Anschließend führt der Prozeß Hervorhebung des Darstellungsblattes den Prozeß "Anzeige des verfügbaren virtuellen Instrumentendienstes" des Darstellungsblattes aus, um einen Dienstausswahlentwurf 370 zu erzeugen, und zeigt ein Graphiksymbol 372 des virtuellen Instrumentendienstes auf dem erzeugten Dienstausswahlentwurf 370 an (Schritt f4). Außerdem führt der Prozeß Hervorhebung des Darstellungsblattes den Prozeß "Anzeige des verfügbaren Testelementnamens" des Darstellungsblattes aus, um einen Testelementauswahlentwurf 360 zu erzeugen, und zeigt ein Graphiksymbol des Testelementnamens auf dem erzeugten Testelementauswahlentwurf 360 an (Schritt f5).

Wenn in diesem Zustand z. B. ein Graphiksymbol 372 der Spitzenwerte auf den Dienstausswahlentwurf 370 gezogen und auf einen Bereich eines Teils des Graphen fallengelassen wird, wird der Prozeß "Spezifiziere virtuellen Instrumentendienst" der Schlitzinformation ausgeführt (Schritt f6). Nach der Ausführung dieses Prozesses wird der Prozeß Dienstaussführung des virtuellen Instrumentendienstes ausgeführt (Schritt f7).

In diesem Beispiel wird der Prozeß Dienstaussführung der Spitzenwerte ausgeführt. Dieser Prozeß findet den Spitzenwert im Bereich 323 eines Abschnitts des Graphen, wobei ein Graphiksymbol, das den Spitzenwert darstellt, an ei-

ner Position angezeigt wird, die dem Spitzenwert zugeordnet ist. Gleichzeitig werden die Werte der X- und Y-Achsen des Spitzenwertes auf dem Graphen angezeigt.

- 5 (3) Prozeßbeispiel der Erzeugung eines neuen Testparameters irgendeines Testelements aus einem spezifizierten Bereich auf der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses und des Startens des neuen Tests mit dem erzeugten Testparameter

### (3.0) Beispiel der GUI-Implementierung

Die Fig. 20 ist ein Beispiel der GUI-Implementierung zum Erzeugen eines neuen Testparameters aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Testergebnisses und zum Starten des Tests.

Wenn der Test ausgeführt worden ist, wird der Testdatenanzeigeentwurf 320 angezeigt, wobei an der Stufe, an der die Testdaten aufgenommen werden, ein Graph im Bereich 322 angezeigt wird, um einen Teil der Testdaten anzuzeigen.

Wenn in einem solchen Zustand der Benutzer einen Bereich 322 eines Abschnitts des Graphen spezifiziert, werden ein Dienstausswahlentwurf 370, der einen Bereich zum Anzeigen des Graphiksymbols des verfügbaren virtuellen Instruments ist, und ein Testelementauswahlentwurf 360 angezeigt, der ein Bereich zum Anzeigen der verfügbaren Testelemente auf dem Testdatenanzeigeentwurf 320 ist. Anschließend werden die Graphiksymbole 362 und 364 der Testelementnamen auf dem Testelementauswahlentwurf 360 angezeigt. Außerdem wird das Graphiksymbol 372 der verfügbaren virtuellen Instrumentendienste auf dem Dienstausswahlentwurf 370 angezeigt.

Wenn das Graphiksymbol des Testelementnamens auf den Testelementauswahlentwurf 360 gezogen und auf den Bereich 323 eines Abschnitts des Graphen fallengelassen wird, wird ein Graphiksymbol 326 des Testelementnamens auf dem Bereich 323 angezeigt. Anschließend werden die folgenden Anzeigeoperationen ausgeführt.

- 40 (a) Wenn der Testelementname keinen durch Vorgabe gestarteten Testgegenstand speichert.

Ein Testgegenstandsauswahlentwurf 380 wird auf dem Basisentwurf der Testaufgabe 300 angezeigt. Anschließend werden die Graphiksymbole 382 und 384 der Testgegenstände, die der ausgewählte Testgegenstand durchführen kann, auf dem Testgegenstandsauswahlentwurf 380 angezeigt. Wenn eines der Graphiksymbole 382 oder 384 des ausgewählten Testgegenstands mittels der Zeigevorrichtung 30 angezeigt wird, wird ein neuer Testdatenanzeigeentwurf 400 erzeugt. Der Test wird mit dem ausgewählten Testgegenstand durchgeführt, wobei dann, wenn der Test abgeschlossen ist, ein Graph 410 auf dem Testdatenanzeigeentwurf 400 angezeigt wird.

- 55 (b) Wenn der Testelementname einen durch Vorgabe gestarteten Testgegenstand speichert

Ein neuer Testdatenanzeigeentwurf 400 wird erzeugt. Der Test wird mit dem ausgewählten Testgegenstand durchgeführt, wobei dann, wenn der Test abgeschlossen ist, ein Graph 410 auf dem Testdatenanzeigeentwurf 400 angezeigt wird. Somit wird ein neuer Testparameter eines beliebigen Testelements aus einem spezifizierten Bereich auf der graphischen Darstellung eines von einem Testelement erhaltenen Testergebnisses erzeugt, wobei der neue Test mit dem erzeugten Testparameter gestartet wird.

(3.1) Ausführungsprozeß zum Erzeugen eines neuen Testparameters eines beliebigen Testelements aus einem spezifizierten Bereich auf der graphischen Darstellung des vom Testelement erhaltenen Testergebnisses, und zum Starten des neuen Tests mit dem erzeugten Testparameter

Die Fig. 21 und 22 sind Flußdiagramme, die eine Operationsprozedur zum Erzeugen eines neuen Testparameters aus einem spezifizierten Bereich auf einer graphischen Darstellung eines Tests und zum Starten eines neuen Testergebnisses zeigen. Die Operationsprozedur der Schritte 11-15 ist die gleiche wie die in Fig. 19 gezeigte Operation, wobei die Beschreibung mit der nächsten Operationsprozedur beginnt.

Wenn das Graphiksymbol des auf den Testelementauswahlentwurf 360 angezeigten Testelementnamens in Fig. 5 gezogen und auf den Bereich 323 eines Abschnitts des Graphen fallengelassen wird, wird der Prozeß Spezifiziere-Testelementname der Schlitzinformation ausgeführt (Schritt g1). Dieser Prozeß Spezifiziere-Testelementname mit der Schlitzinformation führt den Prozeß Starte-Neuen-Test der Wechselwirkungsrichtung aus (Schritt g2).

Der Prozeß Starte-Neuen-Test der Wechselwirkungsrichtung ermittelt zuerst, ob ein Vorgabetestgegenstand in ausgewählten Testelementnamen registriert ist (Schritt g3), wobei der Benutzer des Testsystems dann, falls dies nicht zutrifft, nach dem Testgegenstand gefragt wird (Schritt g4). Dieses Fragen wird durchgeführt mittels Anzeigen eines Graphiksymbols 382 des Testgegenstands, der im Testelement ausgeführt werden kann, auf dem Testgegenstandsauswahlentwurf 380 auf dem Basisentwurf der Testaufgabe 300.

Wenn der Vorgabe-Testgegenstand registriert ist, oder wenn der Benutzer auf das Graphiksymbol 382 des auf dem Testgegenstandsauswahlentwurf 380 angezeigten Testgegenstands oder dergleichen mit der Zeigevorrichtung 30 als Antwort auf die Anfrage zeigt, wird der Prozeß Meßausführung der Wechselwirkungsrichtung ausgeführt (Schritt g5). Der Prozeß Meßausführung der Wechselwirkungsrichtung führt zuerst den Prozeß "erzeuge Testparameter mit Testgegenstand und Schlitzinformation" des virtuellen Instruments aus (Schritt g6), um einen neuen Testparameter, Testdaten und ein Darstellungsblatt zu erzeugen. Die Testklasse wird für diese Erzeugung des Testparameters verwendet. Anschließend führt der Prozeß Meßausführung der Wechselwirkungsrichtung den Prozeß Testausführung des virtuellen Instruments durch, wodurch die Messung mit dem neuen Meßparameter durchgeführt wird.

#### (4) Prozeßbeispiel zur Programmerzeugung

##### (4.0) Beispiel einer GUI für die Programmerzeugung

Die Fig. 23 ist ein Beispiel einer GUI-Implementierung zum Erzeugen und Abarbeiten eines Programms. Wie in der Figur gezeigt, umfaßt der Basisentwurf des Testplans 500 einen Programmelementanzeigentwurf 600 und einen Testplanentwurf 700.

Der Basisentwurf des Testplans 500 ist ein Teil oder das Ganze des Anzeigebereichs auf der graphischen Anzeigevorrichtung zum Erzeugen eines Programms und zum Ausführen des erzeugten Programms. Der Programmelementanzeigentwurf 600 ist ein Anzeigebereich auf dem Basisentwurf des Testplans 500 zum Anzeigen verfügbarer Programmelemente. Der Testplanentwurf 700 ist ein Anzeigebereich auf dem Basisentwurf des Testplans 500 zum Erzeugen eines Programms.

Zu Beginn eines Programms werden auf dem Programmelementanzeigentwurf 600 Programmelement-Graphik-

symbole 610 bis 670 und ein Beginn-Graphiksymbol 710 und ein Ende-Graphiksymbol 750 auf dem Testplanentwurf 700 bereitgestellt (die Graphiksymbole Beginn und Ende werden verbunden und angezeigt).

Um eine Testaufgabe im Programm zu registrieren, wird das Graphiksymbol 350 der Testaufgabe auf den Testaufgabentwurf 330 der Fig. 14 gezogen und auf den Testplanentwurf 700 fallengelassen. Anschließend wird ein Graphiksymbol der Testaufgabe 720 auf dem Testplanentwurf 700 erzeugt.

Zum Auswählen eines Programms, um eine Ausführungsreihenfolge der Testaufgabe zu steuern, wird außerdem ein Graphiksymbol des Testelements auf den Programmelementanzeigentwurf 600 gezogen und auf den Testplanentwurf 700 fallengelassen. Zum Beispiel wird ein Graphiksymbol der Schleife 640 vom Programmelementanzeigentwurf 600 gezogen und auf den Testplanentwurf 700 fallengelassen. Wenn das Graphiksymbol der Schleife 640 auf den Testplanentwurf 700 fallengelassen wird, werden ein Graphiksymbol der Schleife 740 und ein Eingabeabschnitt 742 zum Spezifizieren der Anzahl der Schleifendurchläufe angezeigt.

Um eine Testaufgabe zu spezifizieren, die wiederholt ausgeführt wird, wird das Graphiksymbol der Testaufgabe 350 auf das Graphiksymbol der Schleife 740 fallengelassen. Wenn das Graphiksymbol der Testaufgabe 350 auf das Graphiksymbol der Schleife 740 fallengelassen wird, wird ein Graphiksymbol der Testaufgabe 741 auf dem Graphiksymbol der Schleife 740 erzeugt.

Um das so erzeugte Programm laufen zu lassen, wird mit der Zeigevorrichtung 30 auf das Graphiksymbol Beginn 710 gezeigt. Anschließend wird die Testaufgabe 720 ausgeführt, gefolgt vom Ausführen der Schleife. Wenn die Schleife ausgeführt wird, wird die enthaltene Testaufgabe entsprechend der mit dem Eingabeabschnitt 742 eingegebenen Wiederholungsanzahl ausgeführt (sie wird in diesem Beispiel drei Mal ausgeführt).

Somit kann ein Programm erzeugt werden und das erzeugte Programm abgearbeitet werden.

#### (4.1) Erzeugung des Programms

Die Fig. 24 ist ein Flußdiagramm, die eine Operationsprozedur beim Erzeugen eines Programms zeigt. Es zeigt die Prozedur der Programmerzeugung, die unter Verwendung der Fig. 23 beschrieben worden ist.

Zuerst führt der Testplan den Prozeß initialisiere-Testplan aus (Schritt h1), um den Zustand der GUI zu erzeugen, wenn das Programm startet. Der Prozeß Anbringung der zusammengesetzten Aufgabe, die im Testplan enthalten ist, wird ausgeführt (Schritt h2). Dieser Prozeß wird gestartet durch Ziehen des in Fig. 14 gezeigten Symbols der Testaufgabe 350 auf den Testplanentwurf 700. Als Ergebnis des Prozesses Anbringung der zusammengesetzten Aufgabe, die im Testplan enthalten ist, wird ein Graphiksymbol der Testaufgabe 720 angezeigt.

Anschließend wird der Prozeß Anbringung der zusammengesetzten Aufgabe, die im Testplan enthalten ist, ausgeführt (Schritt h3). Dieser Prozeß wird gestartet durch Ziehen des Symbols der Schleife 640 vom Programmelementanzeigentwurf 600 und Fallenlassen desselben auf dem Testplanentwurf 700. Als Ergebnis des Prozesses Anbringung der im Testplan enthaltenen zusammengesetzten Aufgabe wird eine Schleife erzeugt, wobei ein Graphiksymbol der Schleife 740 angezeigt wird, das ein Graphiksymbol des Eingabeabschnitts 742 enthält.

Anschließend wird der Prozeß Anbringung der in der erzeugten Schleife enthaltenen zusammengesetzten Aufgabe



ausgeführt (Schritt h4). Dieser wird gestartet durch Fallenlassen des Symbols der Testaufgabe auf das Symbol der Schleife 740. Als Ergebnis des Prozesses Anbringung der in der Schleife enthaltenen zusammengesetzten Aufgabe wird das Graphiksymbol der Testaufgabe 741 auf dem Graphiksymbol der Schleife 740 angezeigt.

#### (4.2) Ausführung des Programms

Die Fig. 25 ist ein Flußdiagramm, das eine Operationsprozedur beim Abarbeiten eines Programms zeigt. Es zeigt die Prozedur der Ausführung des unter Verwendung der Fig. 23 beschriebenen Programms.

Zuerst wird der Prozeß Ausführung von Beginn ausgeführt (Schritt j1). Dieser wird gestartet durch Zeigen auf das Graphiksymbol Beginn 710 mittels der Zeigevorrichtung 30. Der Prozeß Ausführung von Beginn erfaßt einen Testplan, der den Beginn zuordnet, und führt den Prozeß Ausführung des erfaßten Testplans aus (Schritt j2). Der Prozeß Ausführung des Testplans führt den Prozeß Ausführung der enthaltenen zusammengesetzten Aufgabe aus. Der Prozeß Ausführung der zusammengesetzten Aufgabe führt die Prozesse Ausführung der Aufgabenkomponenten aus, die in einer Reihenfolge von der Spitze der Matrixdaten ausgehend gehalten werden, welche die Zuordnungsbeziehung halten. Er führt jedoch nicht Beginn aus. In diesem Beispiel ist das Programmelement an der Spitze der Matrixdaten der Beginn. Somit ist die nächste Testaufgabe, die in der zusammengesetzten Aufgabe gehalten wird, die im Testplan enthalten ist, die Testaufgabe, die dem Graphiksymbol 720 zugeordnet ist. Daher führt der Prozeß Ausführung des Testplans den Prozeß Ausführung der Testaufgabe aus, die dem Graphiksymbol 720 zugeordnet ist (Schritt j3).

Außerdem ist ein Programmelement, das die im Testplan enthaltene zusammengesetzte Aufgabe zuordnet, neben der Testaufgabe, die dem Graphiksymbol 720 entspricht, die Schleife. Somit führt der Prozeß Ausführung des Testplans den Prozeß Ausführung der Schleife aus (Schritt j4).

Der Prozeß Ausführung der Schleife führt den Prozeß Ausführung der in der Schleife enthaltenen zusammengesetzten Aufgabe entsprechend der Anzahl der Schleifendurchläufe aus. In diesem Beispiel beträgt die Anzahl der Schleifendurchläufe drei, wobei das von der in der Schleife enthaltenen zusammengesetzten Aufgabe zugeordnete Programmelement nur die Testaufgabe ist, die vom Graphiksymbol 741 dargestellt wird. Somit wird der Prozeß Ausführung der Testaufgabe drei Mal ausgeführt (Schritte j5 bis j7).

Außerdem ist ein Programmelement, das die im Testplan enthaltene zusammengesetzte Aufgabe neben der Schleife zuordnet, das Ende. Somit wird der Prozeß Ausführung vom Ende ausgeführt (Schritt j8). Da die zusammengesetzte Aufgabe, die im Testplan enthalten ist, kein weiteres Programmelement zuordnet, wird der Prozeß Ausführung des Testplans abgeschlossen.

Wie oben beschrieben worden ist, besitzt der GUI-Erzeugungs/Verwaltungsabschnitt 34, der im GUI-Verwaltungsabschnitt 100 des Testsystems gemäß der Ausführung enthalten ist, Funktionen wie die Testaufgabeneinrichtung, die Testparametereinrichtung, die Testdatenspeichereinrichtung, die virtuelle Instrumenteneinrichtung, die Darstellungsblatteinrichtung, die Testklasseneinrichtung, die Schlitzinformationseinrichtung, die virtuelle Instrumentendiensteneinrichtung, die Wechselwirkungseinrichtung und die Testplaneinrichtung und kann (1) eine Operation starten durch Auswählen eines Objekts oder eines Ziels des Tests, (2) Spezifizieren, ob ein Test mit hoher Geschwindigkeit oder mit hoher Genauigkeit ausgeführt wird, (3) einen Dienst starten, der für einen spezifizierten Bereich ausgelegt

ist, ohne ein Programm umzuprogrammieren, wenn bestimmte Koordinaten auf einer graphischen Darstellung eines Testergebnisses spezifiziert werden, (4) einen neuen Testparameter eines Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung des vom Testelement erhaltenen Ergebnisses ohne Umprogrammierung eines Programms erzeugen und einen neuen Test mit dem erzeugten Testparameter starten, und (5) einen neuen Testparameter eines weiteren Testelements aus einem spezifizierten Bereich auf einer graphischen Darstellung des vom Testelement erhaltenen Ergebnisses ohne Umprogrammierung eines Programms erzeugen und einen neuen Test mit dem erzeugten Testparameter starten.

In den Zeichnungen entspricht der Begriff "gehaltenes Objekt" dem Begriff "zugeordnetes Objekt" der Beschreibung, der Begriff "Halten" in den Zeichnungen entspricht dem Begriff "zuordnen" in der Beschreibung, sofern er sich auf Objekte bezieht.

#### Patentansprüche

1. Testsystem zum Durchführen einer Operation einer Anwendung, die eine Testausrüstung steuert, um eine zu testende Vorrichtung zu testen, durch Anzeigen von Bildern, wie z. B. Graphiksymbolen oder Knöpfen, auf einem Bildschirm und durch Auswählen dieser Bilder unter Verwendung einer Zeigevorrichtung (30), wobei das Testsystem umfaßt:

eine Testaufgabeneinrichtung zum Managen eines Testelements und eines Testziels, um die zu testende Vorrichtung zu testen;  
eine Testklasseneinrichtung zum Auswählen eines Testparameters auf der Grundlage des Testziels;  
eine virtuelle Instrumenteneinrichtung zum Durchführen eines Tests mit dem Testelement unter Verwendung des Testparameters, der durch die Testklasseneinrichtung gesetzt ist;  
eine Testdatenspeichereinrichtung zum Speichern der Testdaten, die aus dem von der virtuellen Instrumenteneinrichtung durchgeführten Test erhalten werden;  
eine Testparameterspeichereinrichtung zum Speichern des von der Testklasseneinrichtung gesetzten Parameters; und  
eine Darstellungsblatteinrichtung zum Anzeigen der vom Testsystem aufgenommenen Daten.

2. Testsystem nach Anspruch 1, gekennzeichnet durch eine Schlitzzeineinrichtung, die veranlaßt, daß die Anzeigepositionsinformation des Testergebnisses auf der Darstellungsblatteinrichtung der Eingabeinformation der Zeigevorrichtung (30) entspricht; und  
eine virtuelle Instrumentendiensteneinrichtung, die einen vergebenen Dienst zur Verfügung stellt, wie z. B. eine arithmetische Operation, wenn ein vorgegebener Bereich zum Anzeigen des Testergebnisses mittels der Schlitzzeineinrichtung spezifiziert wird.

3. Testsystem nach Anspruch 1, gekennzeichnet durch eine Schlitzzeineinrichtung, die veranlaßt, daß die Anzeigepositionsinformation des Testergebnisses auf der Darstellungsblatteinrichtung der Eingabeinformation der Zeigevorrichtung (30) entspricht; und  
eine Wechselwirkungseinrichtung zum Durchführen eines neuen Tests durch Kombinieren der virtuellen Instrumenteneinrichtung, der Testklasseneinrichtung und der Schlitzzeineinrichtung.

4. Testsystem nach Anspruch 1, gekennzeichnet durch eine Testplaneinrichtung zum Erzeugen und Abarbeiten eines Programms, wobei das Programm die Steuerung der Ausführungsreihenfolge der Testaufgabenein-

richtung definiert.

5. Verfahren zum Steuern eines Testsystems zur Durchführung einer Operation einer Anwendung, die eine Testausrüstung steuert zum Testen einer zu testenden Vorrichtung, durch Anzeigen von Bildern wie zum Beispiel Graphiksymbolen oder Druckknöpfen auf einem Bildschirm und Auswählen dieser Bilder unter Verwendung einer Zeigevorrichtung (30), wobei das Verfahren umfaßt:

einen ersten Schritt des Anzeigens eines ersten Graphiksymbols, das die zu testende Vorrichtung darstellt, in einem ersten Anzeigebereich, der in einem Anzeigebildschirm enthalten ist;

einen zweiten Schritt des Anzeigens eines zweiten Graphiksymbols, das ein ausgewähltes Testelement darstellt, auf dem ersten Anzeigebereich;

einen dritten Schritt des Verbindens und Anzeigens des ersten Graphiksymbols und des zweiten Graphiksymbols im ersten Anzeigebereich;

einen vierten Schritt des Erzeugens eines Testparameters für das ausgewählte Testelement;

einen fünften Schritt des Durchführens eines vorgegebenen Tests durch Setzen des Testparameters, der im vierten Schritt erzeugt worden ist, und durch aufnehmen der Testdaten vom Testelement; und

einen sechsten Schritt des Anzeigens der im fünften Schritt aufgenommenen Testdaten im ersten Anzeigebereich.

6. Verfahren zum Steuern des Testsystems nach Anspruch 5, ferner vor dem dritten Schritt gekennzeichnet durch

einen siebten Schritt des Anzeigens eines dritten Graphiksymbols im ersten Anzeigebereich, wobei das dritte Graphiksymbol einen oder mehrere Testgegenstände darstellt, die dem ausgewählten Testelement zugeordnet sind; und

einen achten Schritt des Auswählens irgendeines der einen oder mehreren dritten Graphiksymbole, die im siebten Schritt angezeigt werden, wodurch ein Testgegenstand ausgewählt wird.

7. Verfahren zum Steuern des Testsystems nach Anspruch 6, das ferner zwischen dem achten Schritt und dem dritten Schritt gekennzeichnet ist durch

einen neunten Schritt des Anzeigens eines vierten Graphiksymbols im ersten Anzeigebereich, wobei das vierte Graphiksymbol eine Testbedingung darstellt, die im voraus entsprechend dem ausgewählten Testgegenstand vorbereitet worden ist; und

einen zehnten Schritt des Auswählens des im neunten Schritt angezeigten vierten Graphiksymbols, wodurch eine Testbedingung ausgewählt wird.

8. Verfahren zum Steuern des Testsystems nach Anspruch 5, das nach oder parallel zum zweiten Schritt gekennzeichnet ist durch

einen elften Schritt des Anzeigens eines fünften Graphiksymbols im ersten Anzeigebereich, wobei das fünfte Graphiksymbol dem ausgewählten Testelement zugeordnet ist und anhand des zweiten Graphiksymbols identifiziert werden kann; wobei

das zweite Graphiksymbol und das fünfte Graphiksymbol als ein Graphiksymbol verwendet werden, das das Testelement darstellt, das entweder dem Hochgeschwindigkeitstest oder einem hochgenauen Test zugeordnet ist.

9. Verfahren zum Steuern des Testsystems nach Anspruch 8, dadurch gekennzeichnet, daß es anstelle des sechsten Schritts einen zwölften Schritt des Anzeigens der Testdaten, die dem Hochgeschwindigkeitstest oder

dem hochgenauen Test zugeordnet sind, im ersten Anzeigebereich umfaßt.

10. Verfahren zum Steuern des Testsystems nach Anspruch 5, dadurch gekennzeichnet, daß bei diesem die Anzeige der Testdaten im ersten Anzeigebereich erreicht wird durch eine graphische Darstellung in einer vorgegebenen Form; und wobei das Verfahren ferner versehen ist mit

einem dreizehnten Schritt des Überlappens und Anzeigens eines äußeren Rahmens des ausgewählten Bereichs auf der graphischen Darstellung, wenn auf der graphischen Darstellung der im ersten Anzeigebereich angezeigten Testdaten ein Bereich ausgewählt wird.

11. Verfahren zum Steuern des Testsystems nach Anspruch 10, das nach dem dreizehnten Schritt gekennzeichnet ist durch

einen vierzehnten Schritt des Anzeigens eines oder mehrerer sechsten Graphiksymbole im ersten Anzeigebereich, wobei die sechsten Graphiksymbole vorgegebene Dienste darstellen, die auf der graphischen Darstellung implementiert werden können; und einen fünfzehnten Schritt des Ausführens eines entsprechenden Dienstes, wenn irgendeines der sechsten Graphiksymbole ausgewählt wird, und des Anzeigens seines Ergebnisses im ersten Anzeigebereich.

12. Verfahren zum Steuern des Testsystems nach Anspruch 10, das ferner nach dem dreizehnten Schritt gekennzeichnet ist durch

einen sechzehnten Schritt des Anzeigens eines oder mehrerer siebten Graphiksymbole im ersten Anzeigebereich, wobei die siebten Graphiksymbole die Testelemente darstellen, die in der graphischen Darstellung implementiert werden können;

einen siebzehnten Schritt des Auswählens eines der siebten Graphiksymbole, die im sechzehnten Schritt angezeigt werden, durch Auswählen eines Testelements;

einen achtzehnten Schritt des Setzens eines Testbereichs aus den erhaltenen Testdaten auf der Grundlage von Positionskoordination des ausgewählten Bereichs in der graphischen Darstellung;

einen neunzehnten Schritt des Erzeugens eines Testparameters für das im siebzehnten Schritt ausgewählte Testelement;

einen zwanzigsten Schritt des Einsetzens des im neunzehnten Schritt erzeugten Testparameters in das ausgewählte Testelement, um einen vorgegebenen Test durchzuführen, und um neue Testdaten aufzunehmen; und

einem 21. Schritt des Anzeigens der im zwanzigsten Schritt aufgenommenen Testdaten im ersten Anzeigebereich.

13. Verfahren zum Steuern des Testsystems nach Anspruch 12, das ferner nach dem sechzehnten Schritt gekennzeichnet ist durch

einen 22. Schritt des Anzeigens eines oder mehrerer achten Graphiksymbole im ersten Anzeigebereich, wobei die achten Graphiksymbole einen oder mehrere Testgegenstände darstellen, die dem ausgewählten Testelement zugeordnet sind; und

einen 23. Schritt des Auswählens eines der achten Graphiksymbole, die im 22. Schritt angezeigt werden, wodurch ein Testgegenstand ausgewählt wird; und wobei das System ferner anstelle des neunzehnten Schritts einen 24. Schritt des Erzeugens eines Testparameters für den im siebzehnten Schritt auf der Grundlage des im 23. Schritt ausgewählten Testgegenstand und den im achtzehnten Schritt ausgewählten Testbereich umfaßt.

14. Verfahren zum Steuern des Testsystems nach Anspruch 5, gekennzeichnet durch  
einen 25. Schritt des Anzeigens eines neunten Graphik-  
symbols, das ein oder mehrere Programmelemente dar-  
stellt, in einem zweiten Anzeigebereich, der im Anzei-  
gebildschirm oder im ersten Anzeigebereich enthalten  
ist;  
einen 26. Schritt des Anzeigens eines zehnten Graphik-  
symbols, das den Start eines Tests darstellt, und eines  
elften Graphiksymbols, das das Ende des Tests dar-  
stellt, im zweiten Anzeigebereich, und des Verbindens  
und Anzeigens dieser zehnten und elften Graphiksym-  
bole im zweiten Anzeigebereich;  
einen 27. Schritt des Auswählens eines der neunten  
Graphiksymbole, die im zweiten Anzeigebereich oder  
im ersten Anzeigebereich angezeigt werden, wodurch  
ein Programmelement ausgewählt wird;  
einen 28. Schritt des Auswählens einer Position zum  
Anzeigen eines zwölften Graphiksymbols zwischen  
den zehnten und elften Graphiksymbolen, wobei das  
zwölfte Graphiksymbol dem Programmelement ent-  
spricht, das im 27. Schritt ausgewählt worden ist;  
einen 29. Schritt des Löschens der im 26. Schritt erhal-  
tenen Verbindung zwischen den zehnten und elften  
Graphiksymbolen;  
einen 30. Schritt des Anzeigens des 12. Graphiksym-  
bols an der vorgegebenen Position auf dem zweiten  
Anzeigebereich, die im 28. Schritt gewählt worden ist;  
einen 31. Schritt des Ausrichtens der zehnten und elf-  
ten Graphiksymbole und des einen oder der mehreren  
zwölften Graphiksymbole zwischen diesen nach dem  
30. Schritt; und  
einen 32. Schritt des Verbindens und Anzeigens zwi-  
schen den zehnten, elften und zwölften Graphiksymbo-  
len in einer vorgegebenen Reihenfolge nach dem 31.  
Schritt.
15. Verfahren zum Steuern des Testsystems nach An-  
spruch 14, dadurch gekennzeichnet, daß die Reihen-  
folge der zwölften Graphiksymbole zwischen dem  
zehnten Graphiksymbol, das den Start des Tests dar-  
stellt, und dem elften Graphiksymbol, das das Ende des  
Tests darstellt, ersetzt werden kann.
16. Verfahren zum Steuern des Testsystems nach An-  
spruch 14, dadurch gekennzeichnet, daß das Pro-  
grammelement ein zusammengesetztes Programmele-  
ment enthält.
17. Verfahren zum Steuern des Testsystems nach An-  
spruch 14, dadurch gekennzeichnet, daß dann, wenn  
das zehnte Graphiksymbol, das den Start eines Tests  
darstellt, ausgewählt ist, die Programmelemente, die  
dem zwölften Graphiksymbolen zugeordnet sind, in ei-  
ner Reihenfolge vom nächsten bis zum zehnten Gra-  
phiksymbol ausgeführt werden.

- Leerseite -

FIG. 1

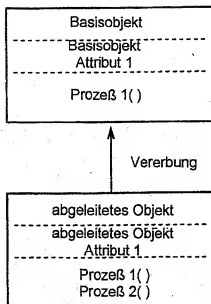


FIG. 2

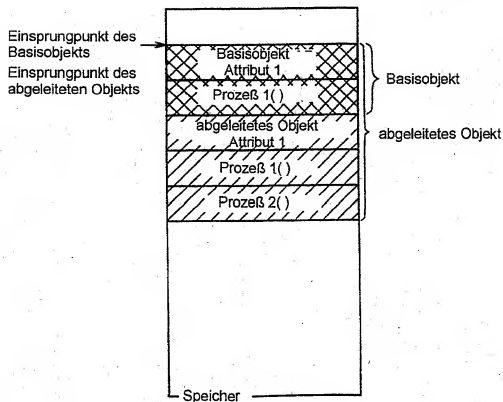


FIG. 3

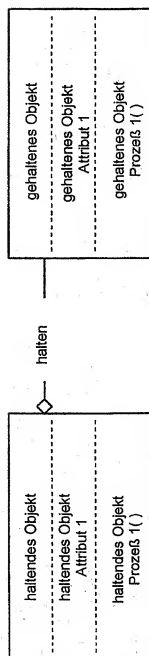


FIG. 4

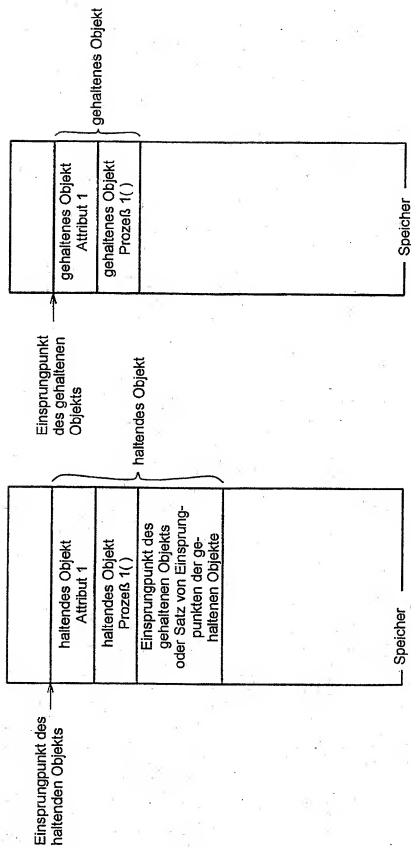


FIG. 5

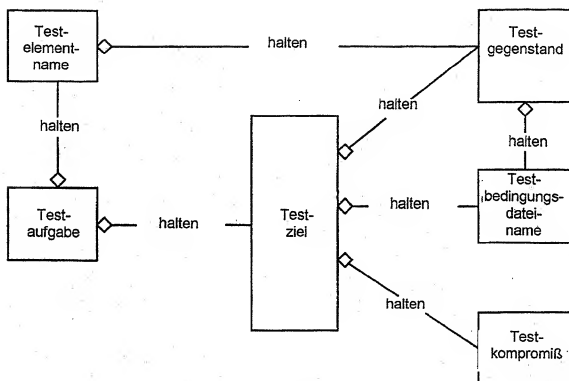




FIG. 6

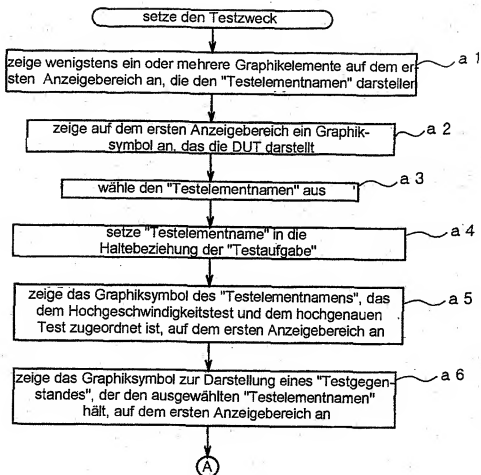


FIG. 7

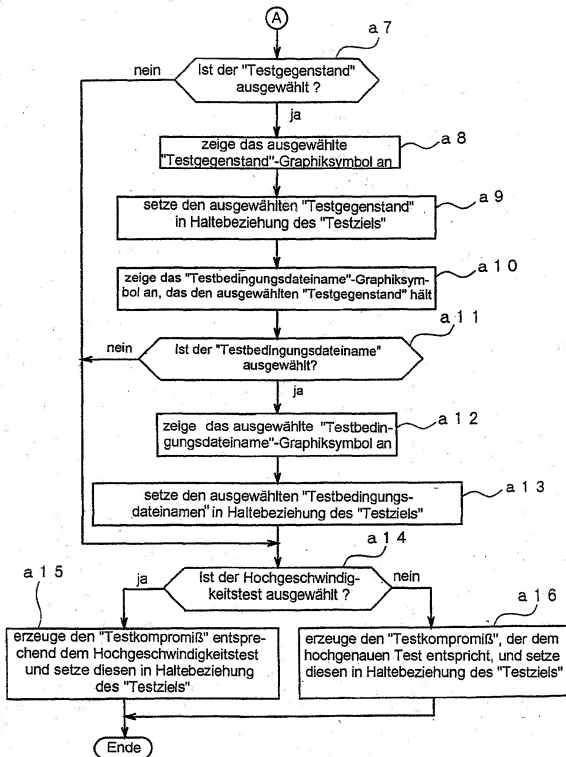




FIG. 9

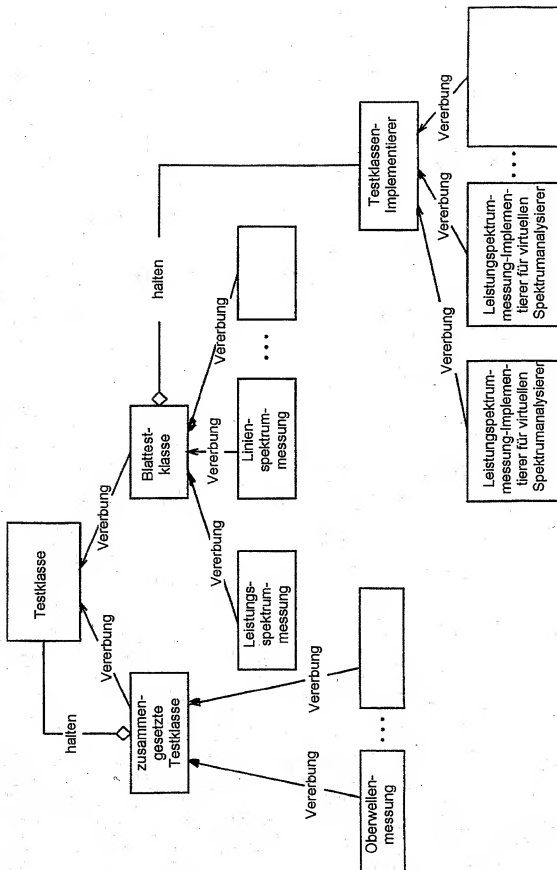


FIG. 10

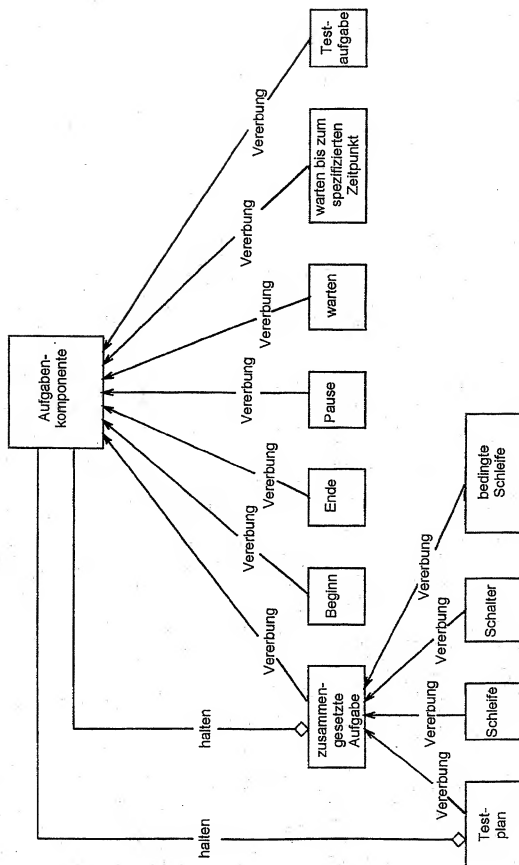


FIG. 11

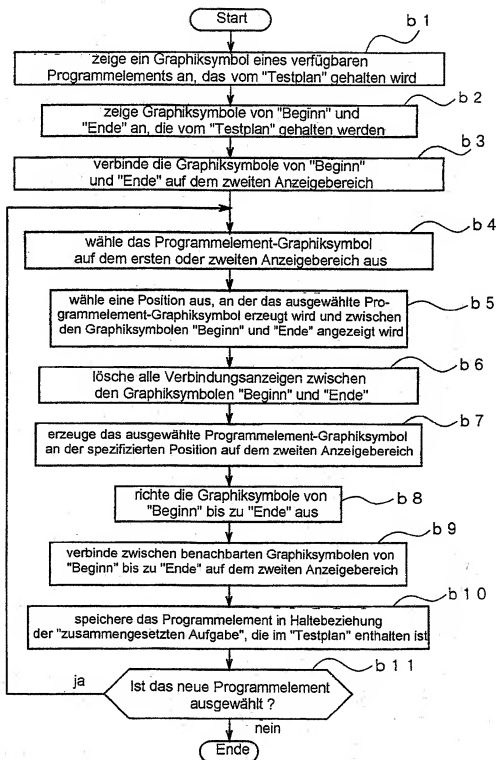


FIG. 12

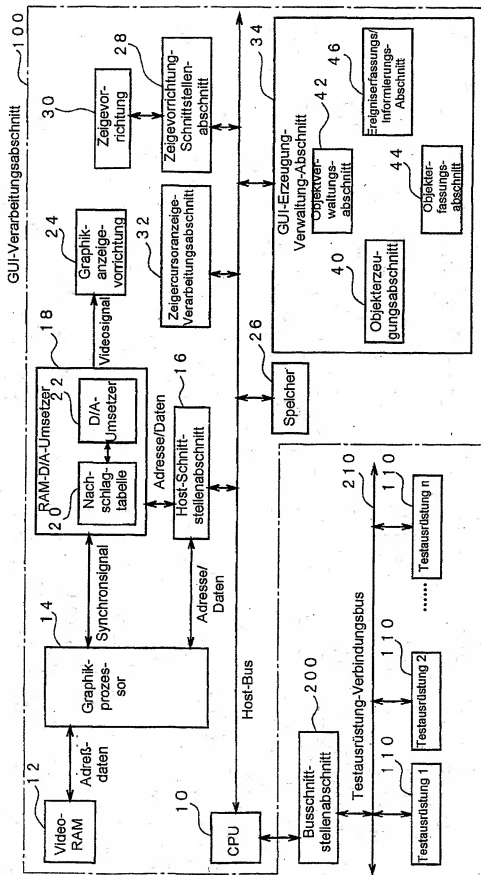


FIG. 13

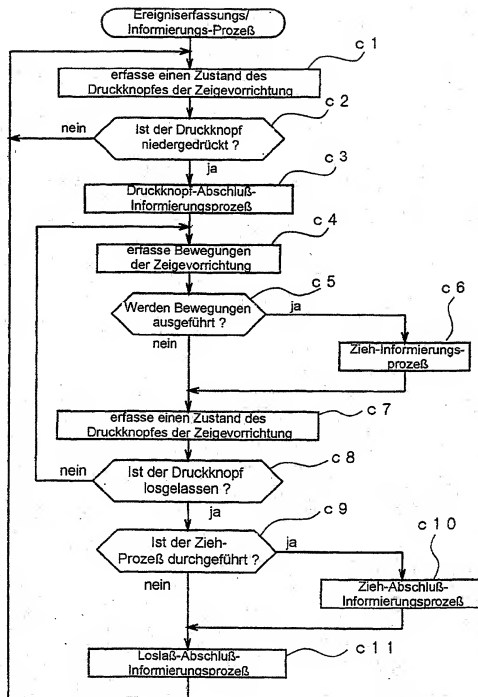




FIG. 14

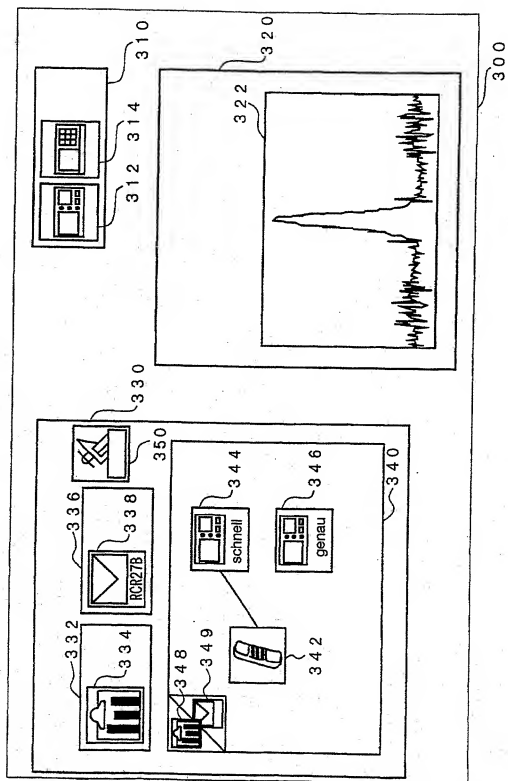


FIG. 15

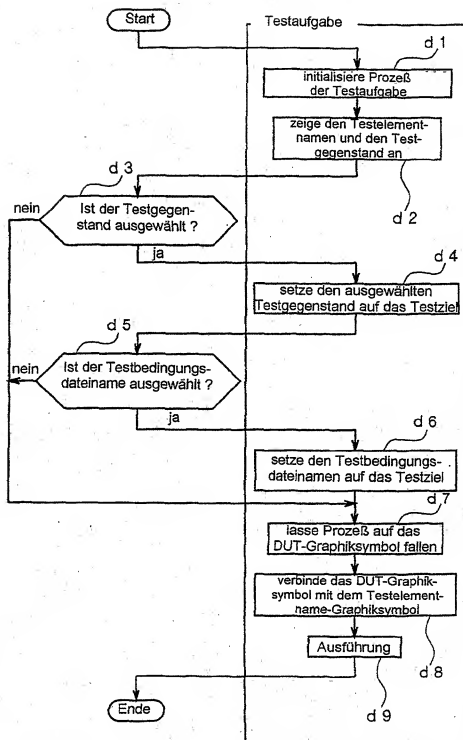


FIG. 16

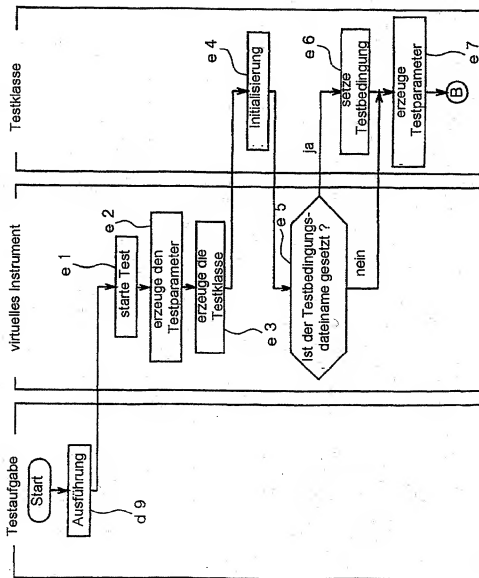


FIG. 17

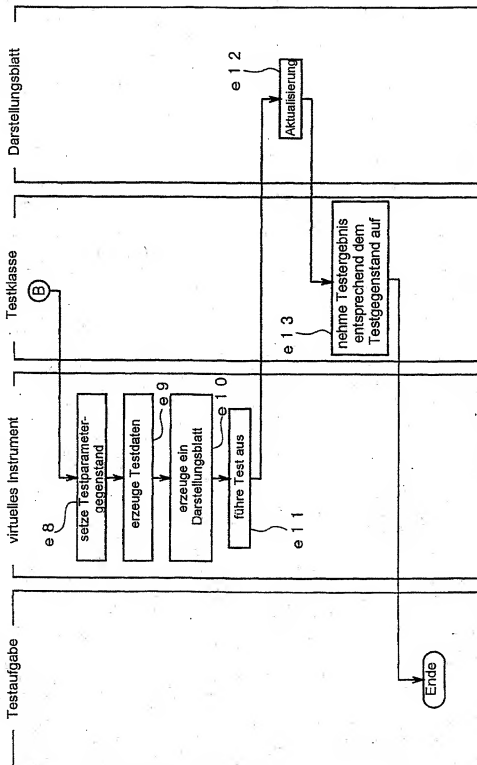


FIG. 18

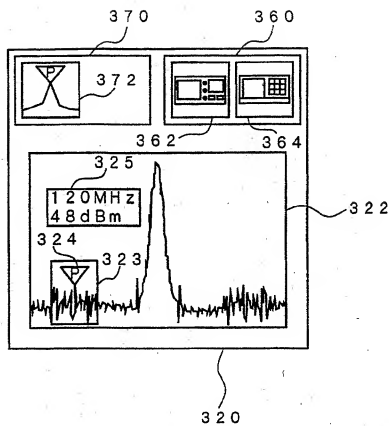


FIG. 19

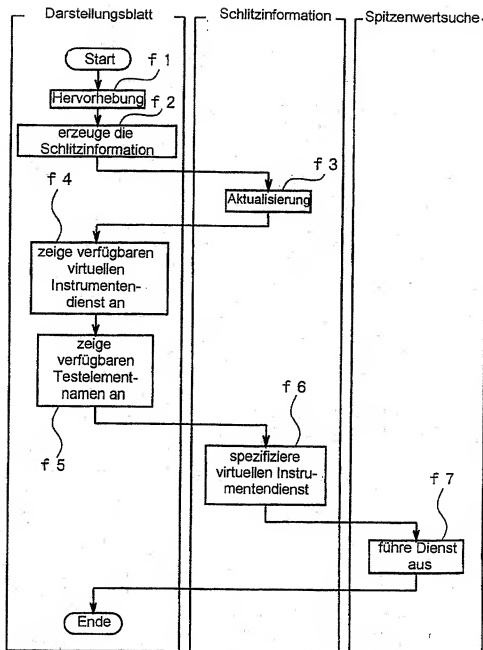


FIG. 20

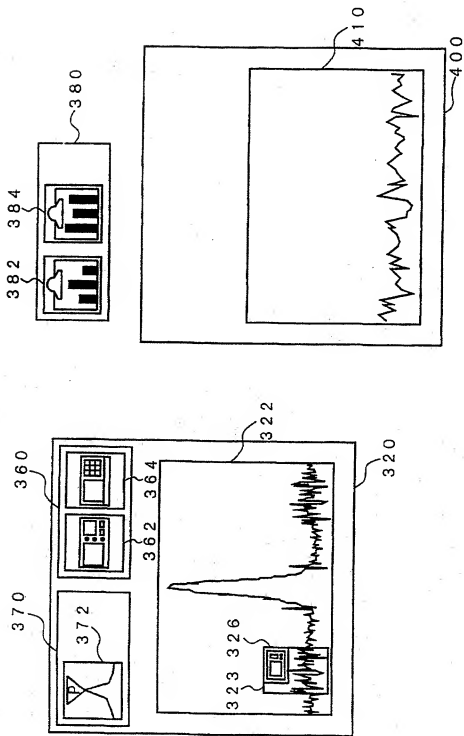


FIG. 21

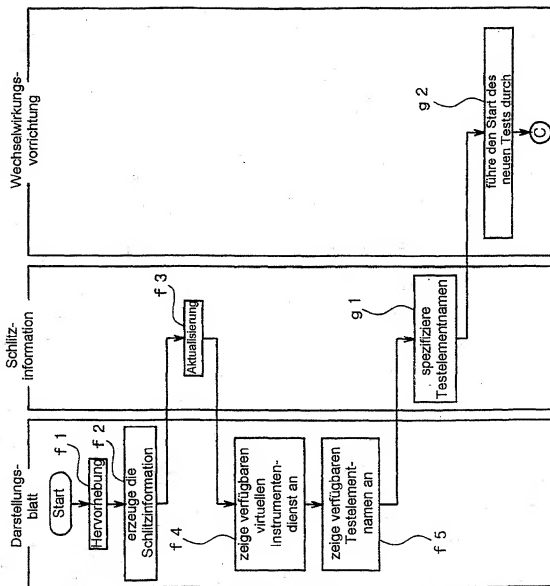




FIG. 22

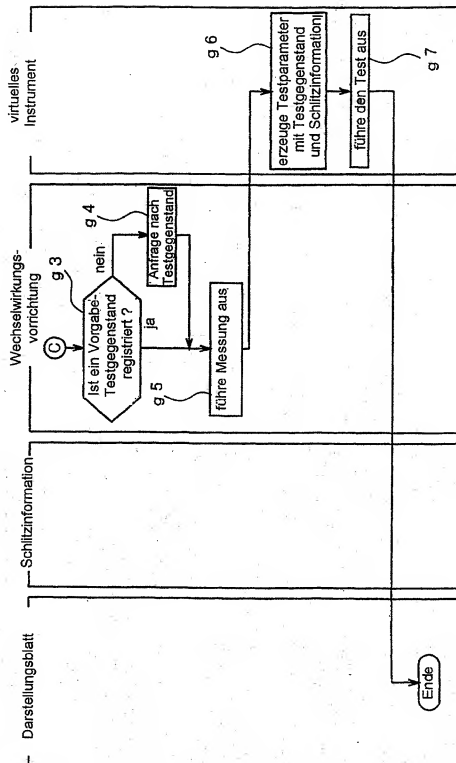


FIG. 23

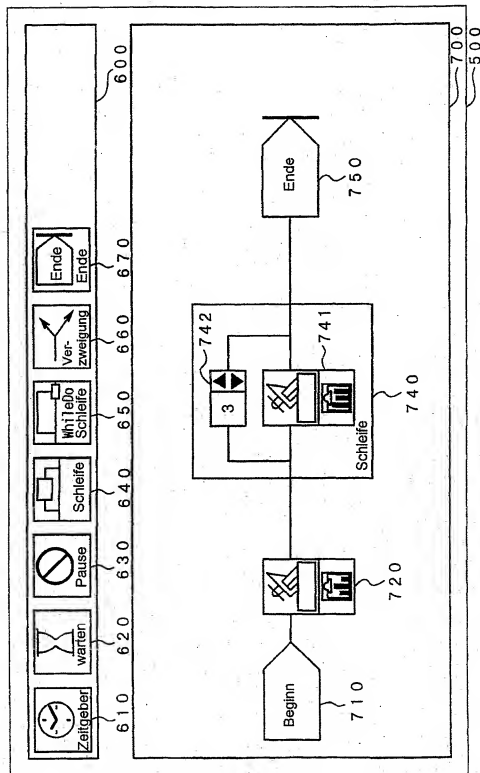


FIG. 24

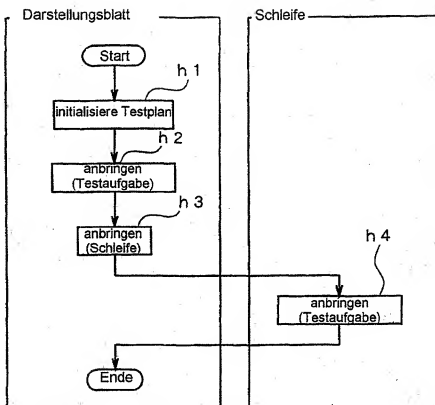


FIG. 25

